12 – Penapisan Citra dalam Ranah Frekuensi

IF4073 Pemrosesan Citra Digital

Oleh: Rinaldi Munir

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung
2024

Penapisan dalam ranah frekuensi

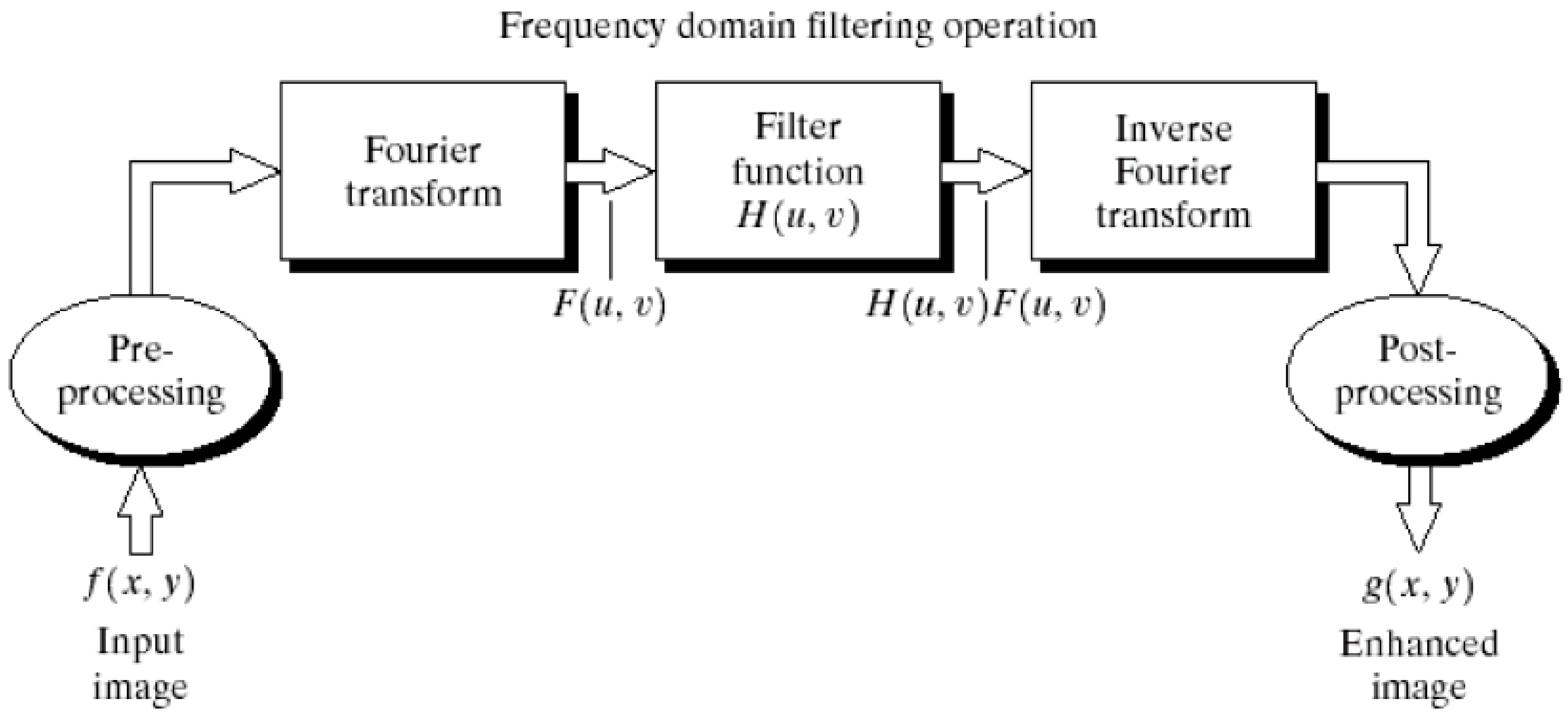


FIGURE 4.5 Basic steps for filtering in the frequency domain.

Hubungan antara operasi penapisan dalam ranah spasial dan ranah frekuensi:

$g(x,y) = h(x,y)*f(x,y) \leftrightarrow G(u,v) = H(u,v)F(u,v)$

dan sebaliknya: $g(x,y) = h(x,y)f(x,y) \leftrightarrow G(u,v) = H(u,v) * F(u,v)$

Catatan: ukuran matriks H(u,v) dan F(u,v) harus sama

```
>> A = magic(5)

A =
    17    24     1     8    15
    23     5     7    14    16
     4     6    13    20    22
    10    12    19    21     3
    11    18    25     2     9
```

```
>> Af = fft2(A)

Af =
   1.0e+02 *
   3.2500 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i
  -0.0000 - 0.0000i   1.0113 - 0.3286i  -0.2023 - 0.0657i   0.0000 + 0.0000i   0.0000 - 0.0000i
   0.0000 + 0.0000i   0.0000 - 0.0000i  -0.3863 - 0.5317i  -0.0000 + 0.0000i   0.0773 - 0.1063i
   0.0000 + 0.0000i   0.0773 + 0.1063i  -0.0000 + 0.0000i  -0.3863 + 0.5317i   0.0000 + 0.0000i
  -0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 - 0.0000i  -0.2023 + 0.0657i   1.0113 + 0.3286i
```

```
>> B = ones(5)

B =
     1     1     1     1     1
     1     1     1     1     1
     1     1     1     1     1
     1     1     1     1     1
     1     1     1     1     1
```

```
>> Bf = fft2(B)

Bf =
    25     0     0     0     0
     0     0     0     0     0
     0     0     0     0     0
     0     0     0     0     0
     0     0     0     0     0
```

```
>> C = Af .* Bf

C =
        8125           0           0           0           0
           0           0           0           0           0
           0           0           0           0           0
           0           0           0           0           0
           0           0           0           0           0
```

```
>> C = Af .* Bf

C =
     8125      0      0      0      0
        0      0      0      0      0
        0      0      0      0      0
        0      0      0      0      0
        0      0      0      0      0
```

```
>> D = ifft2(C)

D =
   325   325   325   325   325
   325   325   325   325   325
   325   325   325   325   325
   325   325   325   325   325
   325   325   325   325   325
```

- Perhatikan bahwa penapisan dalam ranah spasial dan dalam ranah frekuensi keduanya berkoresponden.

$$f(x,y) * h(x,y) \Leftrightarrow H(u,v)F(u,v)$$

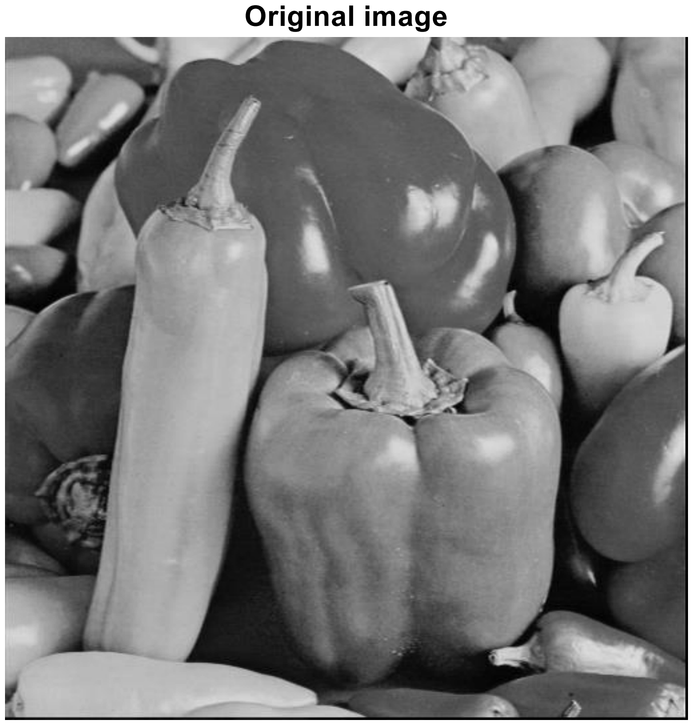
- Contoh: citra *pepper* akan dilakukan *sharpening* pada ranah spasial dan ranah frekuensi. Penapis yang digunakan adalah

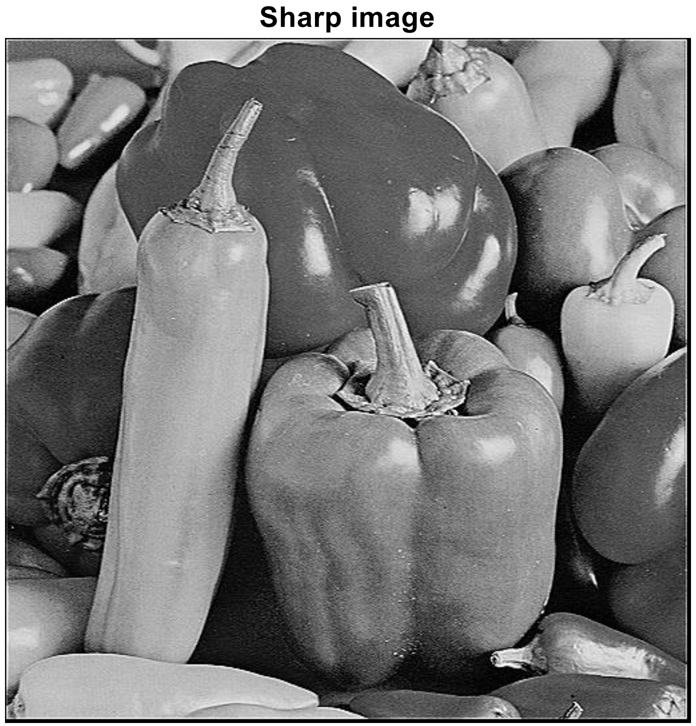
$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$
$$\sum = 1$$

- Citra *f* dan penapis *h* keduanya ditransformasi ke dalam ranah frekuensi
- Sebelum ditransformasi, penapis *h* harus disamakana ukurannya dengan citra f dengan proses padding

- Penapisan dalam ranah spasial (penajaman tepi):

```
f=imread('lada-gray.bmp');
imshow(f), title('Original image');
h = [0 -1 0; -1 5 -1; 0 -1 0];
fsharp = uint8(convn(double(f), double(h)));
figure, imshow(fsharp), title('Sharp image');
```



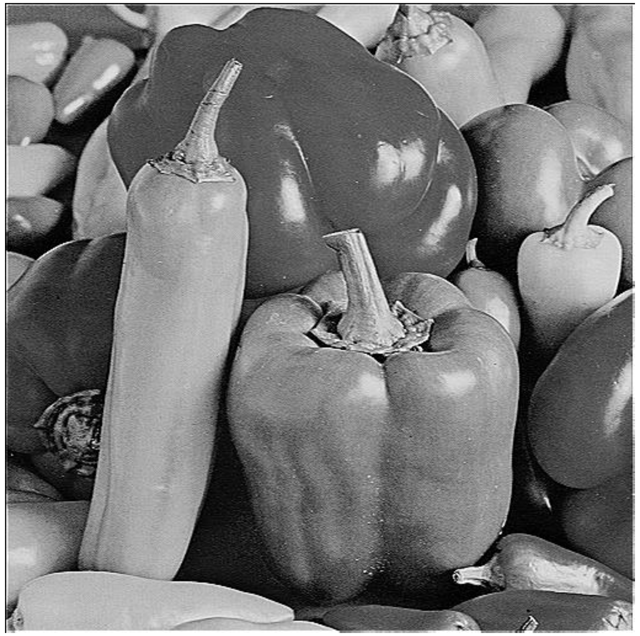


- Penapisan dalam ranah frekuensi:

```
[f, map] = imread('lada-gray.bmp');
imshow(f, map), title('Original image');
h = [0 -1 0; -1 5 -1; 0 -1 0];
[M,N] = size(f);
F = fft2(double(f));
H = fft2(double(h),M,N);
F2 = H.*F;
f2 = ifft2(F2);
g = real(f2);
figure, imshow(g, map), title('Sharp image');
```

Original image

Sharp image



Lowpass filter (LPF) dalam ranah frekuensi

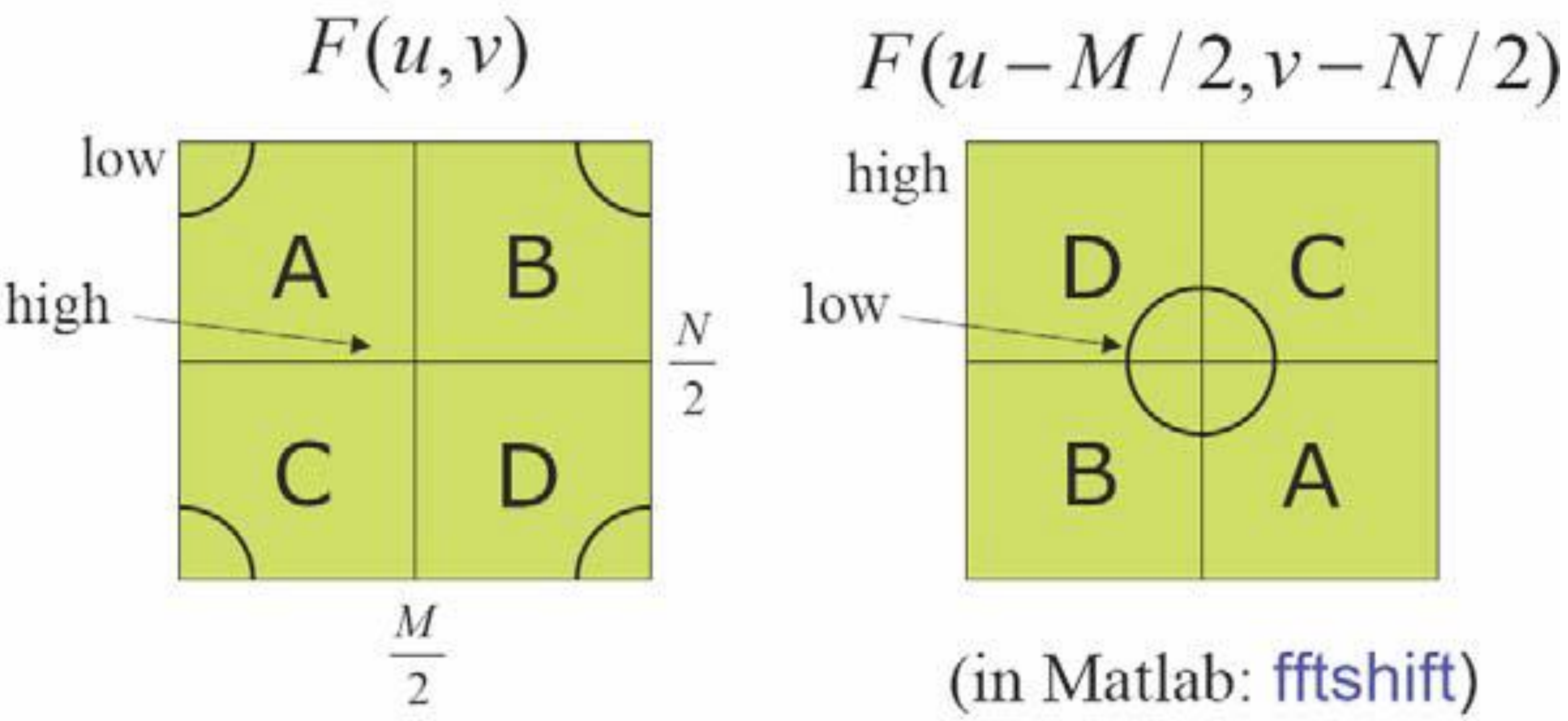
- Penapis lolos rendah (*lowpass filter*) bertujuan menekan komponen berfrekuensi tinggi dan membiarkan komponen berfrekuensi rendah relatif tidak berubah.

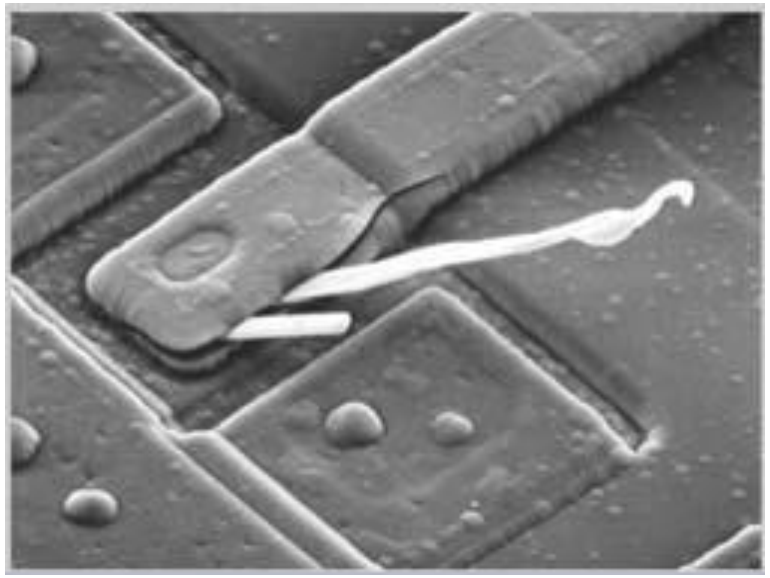
- Menghasilkan efek *blurring* (atau *smoothed image*)

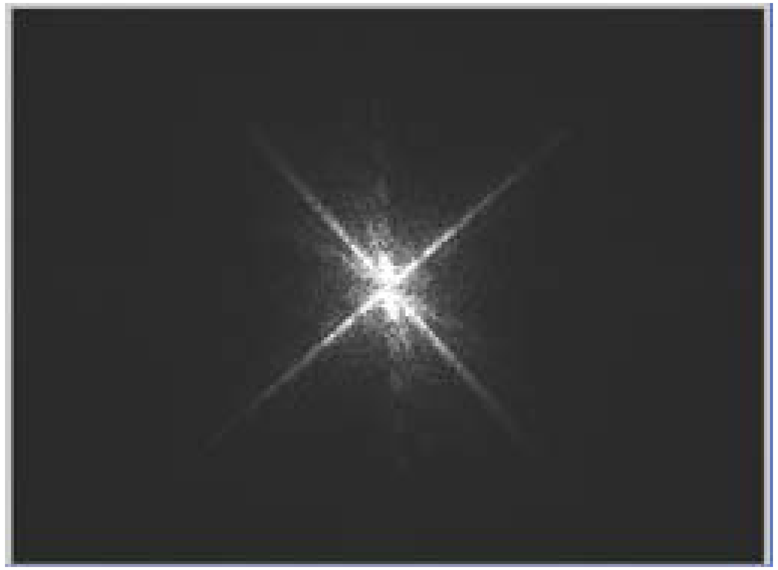
- Tiga buah LPF yang utama:
    1. *Ideal lowpass filter* (ILPF)
    2. *Gaussian lowpass filter* (GLPF)
    3. *Butterworth lowpass filter* (BLPF)

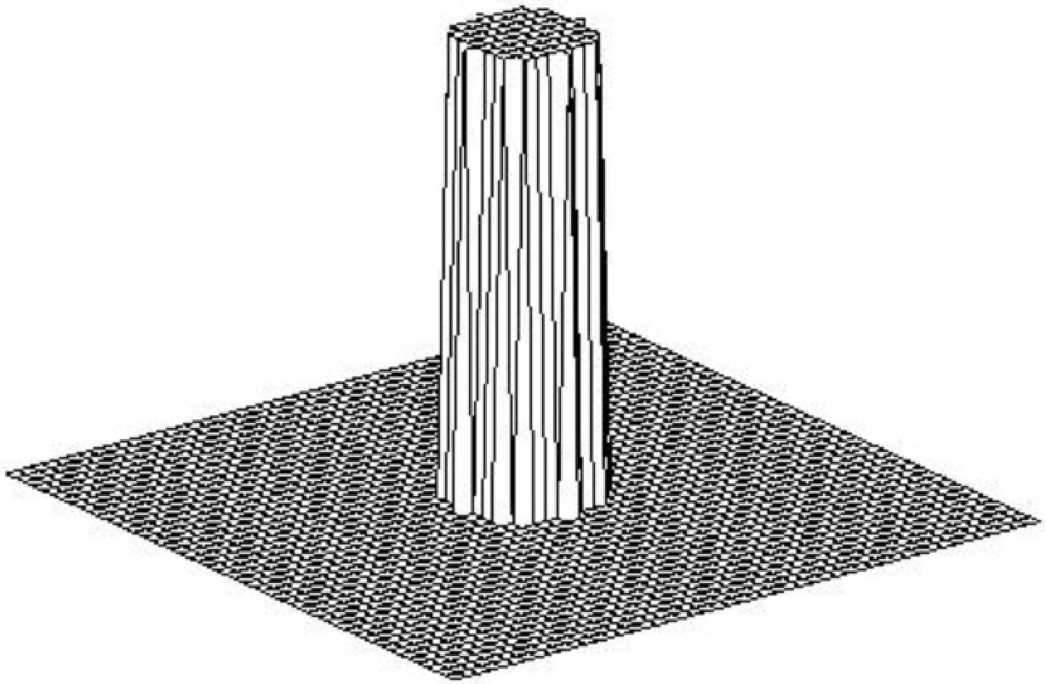
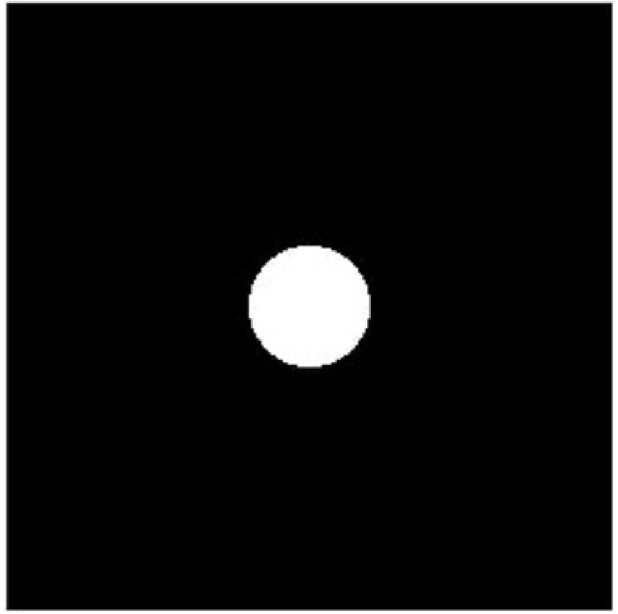
Tinjau Kembali Fourier Transform

$$f(x,y)(-1)^{x+y} \longleftrightarrow F(u-M/2, v-N/2)$$







Lowpass Filter	Formula	Mesh	Image
Ideal	$H(u,v) = \begin{cases} 1 & \text{if } D(u,v) \le D_0 \\ 0 & \text{if } D(u,v) > D_0 \end{cases}$		

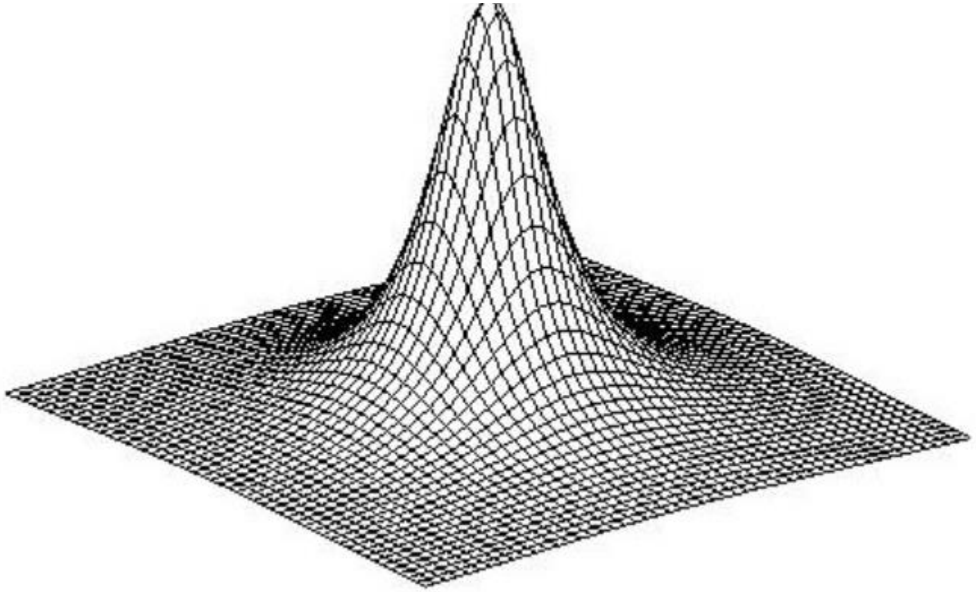
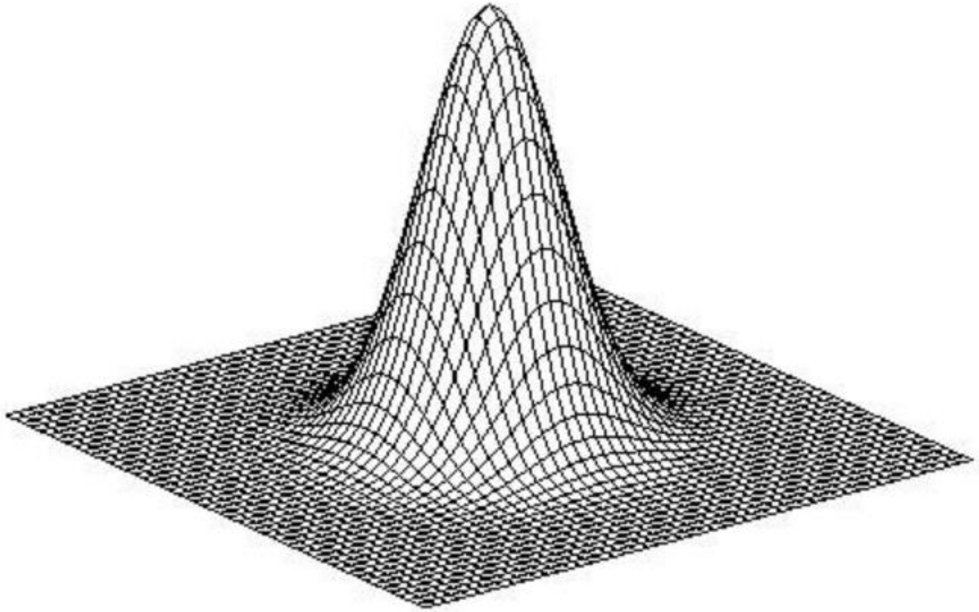
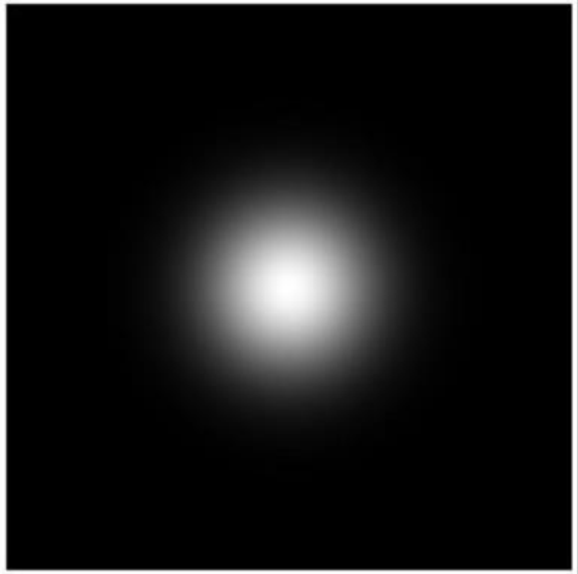
Where $D(u,v)$ is the distance from point (u,v) to the center of the frequency rectangle

If the center is at $(M/2, N/2)$

$$D(u,v) = \sqrt{(u - M/2)^2 + (v - N/2)^2}$$

- $D0$: cutoff frequency

Sumber: https://www.cs.uregina.ca/Links/class-info/425-nova/Lab5/index.html

Butterworth	$H(u,v) = \frac{1}{1+\left[D(u,v)/D_0\right]^{2n}}$		
Gaussian	$H(u,v) = e^{-D^2(u,v)/2D_0^2}$		

Pertimbangan nilai frekuensi *cutoff* (D0)

- Frekuensi *cutoff* D0 dari ILPF ideal menentukan banyaknya komponen frekuensi yang dilewatkan oleh penapis (filter).

- Makin kecil nilai D0, makin banyak komponen gambar yang dihilangkan oleh filter (karena banyak 0 di dalam H(u,v)).

- Secara umum, nilai D0 dipilih sedemikian rupa sehingga sebagian besar komponen yang diinginkan dilewatkan, sementara sebagian besar komponen yang tidak diinginkan dihilangkan.

Langkah-Langkah penapisan dalam ranah frekuensi (dengan Matlab):

1. Tentukan parameter *padding*, biasanya untuk citra f(x,y) berukuran M x N, umumnya parameter *padding* P dan Q adalah P = 2M and Q = 2N.
2. Bentuklah citra *padding* fp(x,y) berukuran P X Q dengan menambahkan pixel-pixel bernilai nol pada f(x, y).
3. Lakukan transformasi Fourier pada fp(x, y)

   Kode Matlab: `F = fft2(fp);`

   Catatan: Langkah 2 dan 3 dapat digabung menjadi sbb: `F = fft2(f, P, Q);`
4. Bangkitkan fungsi penapis H berukuran P x Q, lakukan pemusa

   Catatan: jika penapis diambil dari matriks pada ranah spasial, transformasi penapis dengan Fourier transform.
5. Kalikan F dengan H

   Kode Matlab: `G=H.*F;`
6. Ambil bagian real dari inverse FFT of G:

   Kode Matlab: `g=real(ifft2(G));`
7. Potong bagian kiri atas sehingga menjadi berukuran citra semula

```
g=g(1:size(f,1), 1:size(f,2));
```

Catatan:

Langkah *padding* (1 dan 2) adalah opsional, bertujuan agar citra f(x,y) dan penapis H(u,v) memiliki ukuran yang sama

original image

M

N

padded image

P

Q

Ideal Low Pass Filter (ILPF)

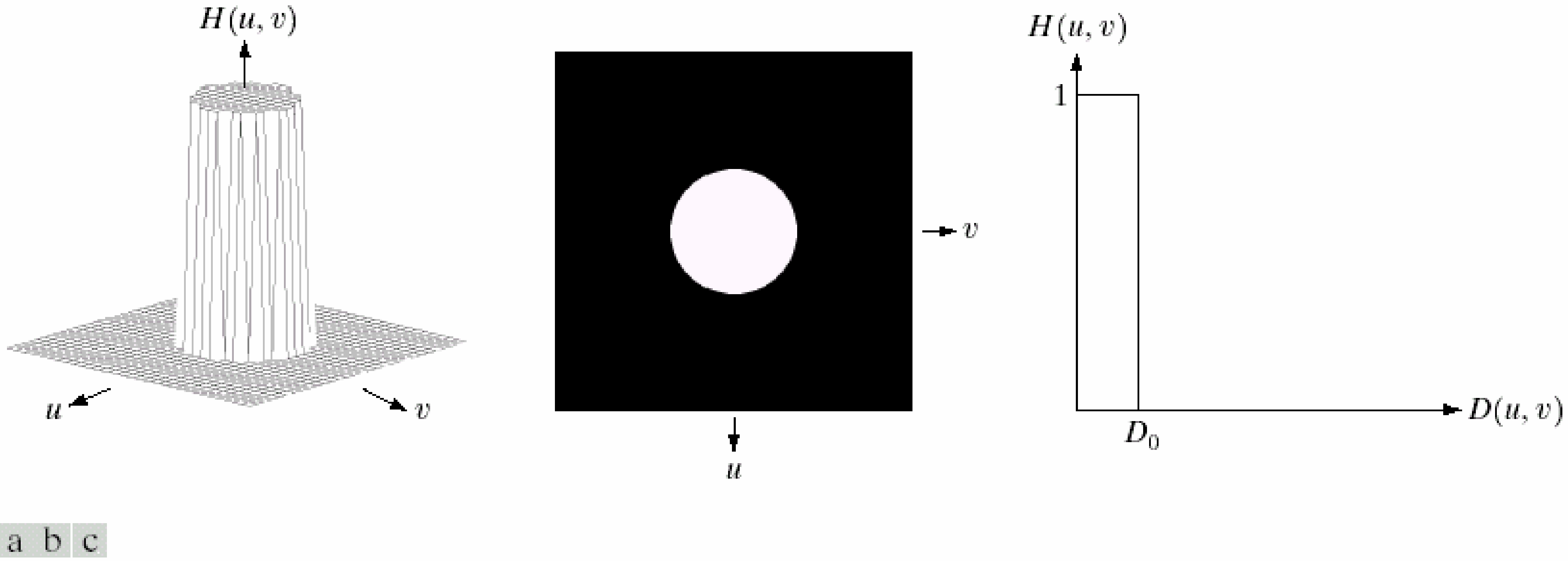


FIGURE 4.10 (a) Perspective plot of an ideal lowpass filter transfer function. (b) Filter displayed as an image. (c) Filter radial cross section.

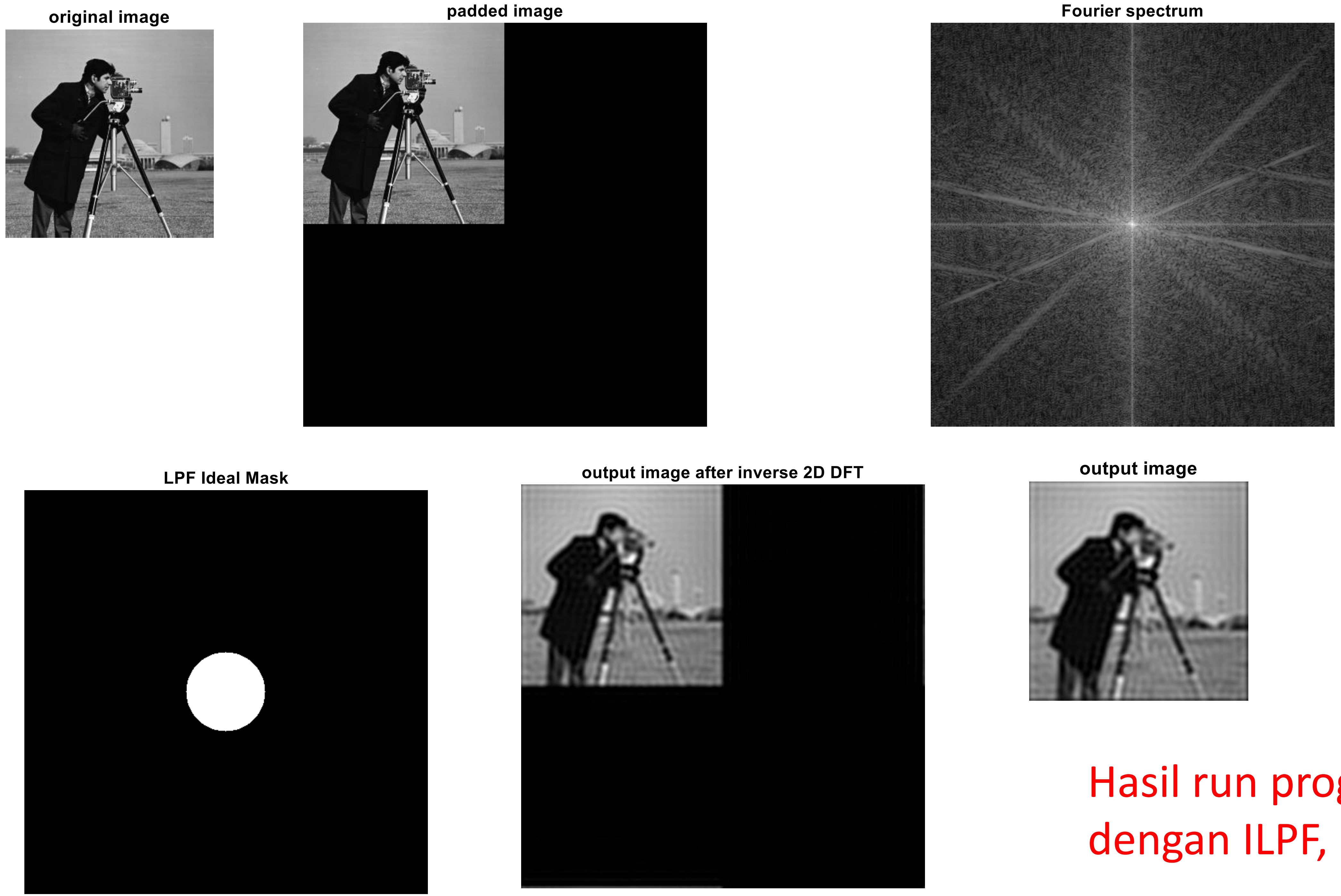
```
% Penapisan dalam ranah frekuensi dengan Ideal Lowpass Filter (ILPF)
f = imread('cameraman.bmp');
imshow(f);
[M,N] = size(f);
%Step 1: Tentukan parameter padding, biasanya untuk citra f(x,y)
%        berukuran M x N, parameter padding P dan Q biasanya P = 2M and Q = 2N.
P = 2*M;
Q = 2*N;

%Step 2: Bentuklah citra padding fp(x,y) berukuran  P X Q dengan
% menambahkan pixel-pixel bernilai nol pada f(x, y).
f = im2double(f);
for i = 1:P
    for j = 1:Q
        if i <= M && j<= N
            fp(i,j) = f(i,j);
        else
            fp(i,j) = 0;
        end
    end
end
imshow(f);title('original image');
figure; imshow(fp);title('padded image');
```

```
%Step 3: Lakukan transformasi Fourier pada fp(x, y) dan tampilkan Fourier
Spectrum
F = fftshift(fft2(fp)); % move the origin of the transform to the center of
the frequency rectangle
S2 = log(1+abs(F)); % use abs to compute the magnitude (handling imaginary)
and use log to brighten display
figure, imshow(S2,[]); title('Fourier spectrum');

%Step 4: Bangkitkan fungsi penapis H berukuran P x Q, misalkan penapis
%yang digunakan adalah Ideal Lowpass Filter (ILPF)

D0 = 50;    % cut-off frequency
% Set up range of variables.
u = 0:(P-1);
v = 0:(Q-1);

% Compute the indices for use in meshgrid
idx = find(u > P/2);
u(idx) = u(idx) - P;
idy = find(v > Q/2);
v(idy) = v(idy) - Q;
```

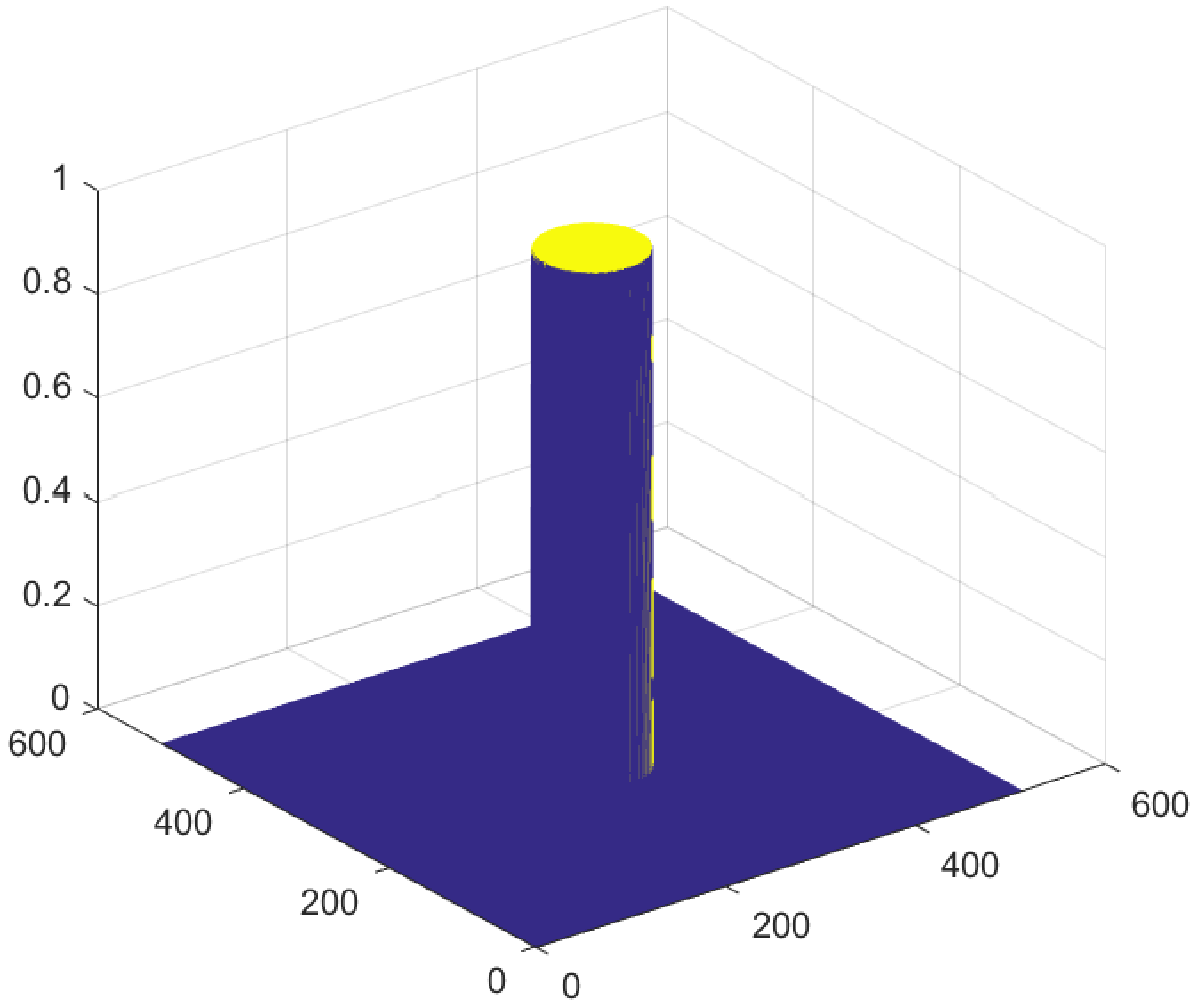
```
% Compute the meshgrid arrays
[V, U] = meshgrid(v, u);
D = sqrt(U.^2 + V.^2);

H = double(D <=D0);
H = fftshift(H); figure;imshow(H);title('LPF Ideal Mask');
figure, mesh(H);

%Step 5: Kalikan F dengan H
G = H.*F;
G1 = ifftshift(G);

%Step 6: Ambil bagian real dari inverse FFT of G1:
G2 = real(ifft2(G1)); % apply the inverse, discrete Fourier transform
figure; imshow(G2);title('output image after inverse 2D DFT');

%Step 7: Potong bagian kiri atas sehingga menjadi berukuran citra semula
G2 = G2(1:M, 1:N); % Resize the image to undo padding
figure, imshow(G2); title('output image');
```

Sumber: Program di atas merupakan modifikasi dari program yang diambil dari sini:

1. https://www.cs.uregina.ca/Links/class-info/425-nova/Lab5/index.html
2. https://www.mathworks.com/matlabcentral/fileexchange/53250-filtering-of-an-image-in-frequency-domain



Hasil run program dengan ILPF, D0 = 50



mesh(H)

Catatan: *Padding* tidak selalu harus dilakukan (opsional), program di atas tetap benar jika tidak dilakukan *padding*, yaitu P = M, dan Q = N.

Semula:

```
% Penapisan dalam ranah frekuensi
% dengan Ideal Lowpass Filter (ILPF)
f=imread('camera.bmp');
imshow(f);
[M,N] = size(f);
% Step 1: Tentukan parameter padding,
% biasanya untuk citra f(x,y)
% berukuran M x N, parameter padding P dan Q
% adalah P = 2M and Q = 2N.
P = 2*M;
Q = 2*N;
```

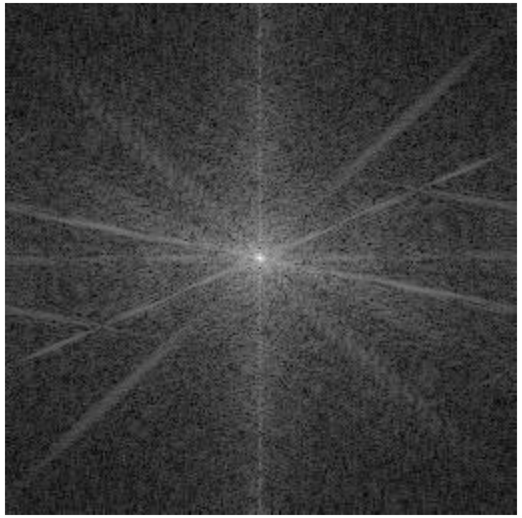
Menjadi

```
% Penapisan dalam ranah frekuensi
% dengan Ideal Lowpass Filter (ILPF)
f=imread('camera.bmp');
imshow(f);
[M,N] = size(f);
%Step 1: Tentukan parameter padding,
% biasanya untuk citra f(x,y)
% berukuran M x N, parameter padding
% P dan Q adalah P = 2M and Q = 2N.
P = M;
Q = N;
```

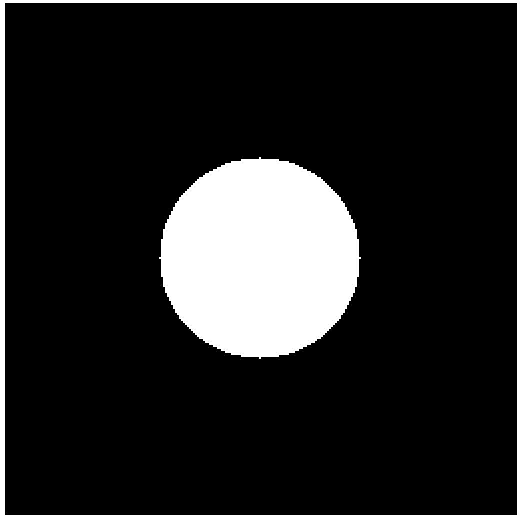
Hasil run program:

original image

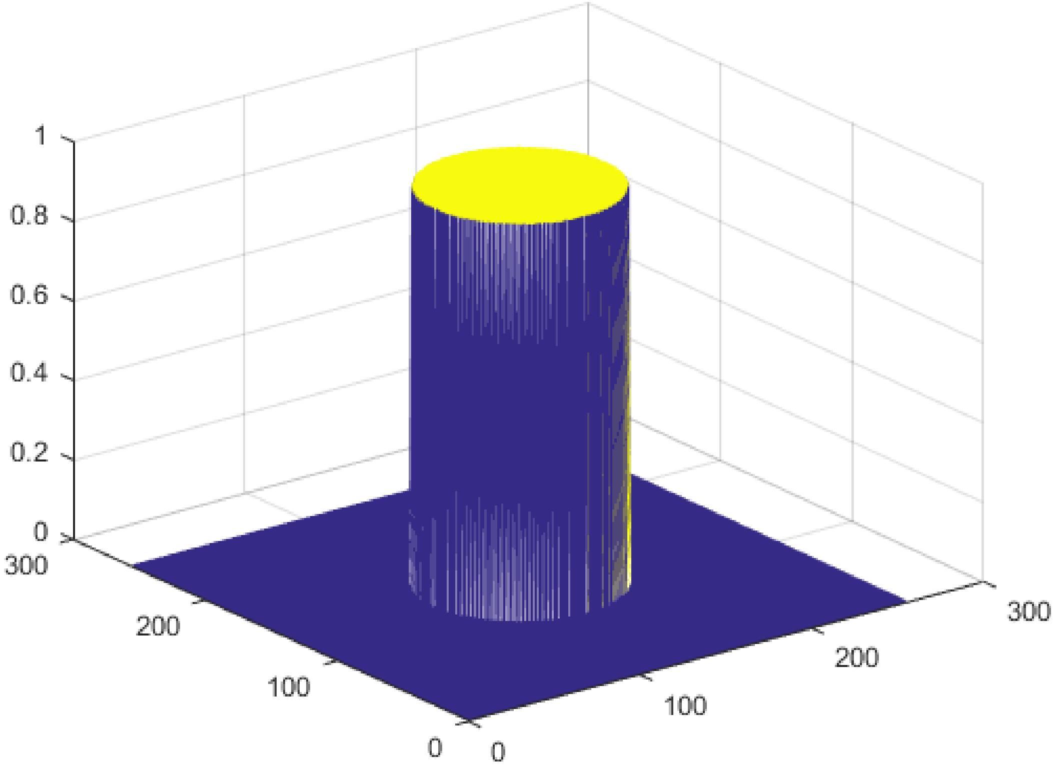
Fourier spectrum



LPF Ideal Mask



output image



Kelemahan hasil ILPF

- Menimbulkan efek bergetar (*ringing*) pada citra *blur*, sebagai karakteristik ILPF
- Ini akibat dari diskontinuitas pada fungsi penapis

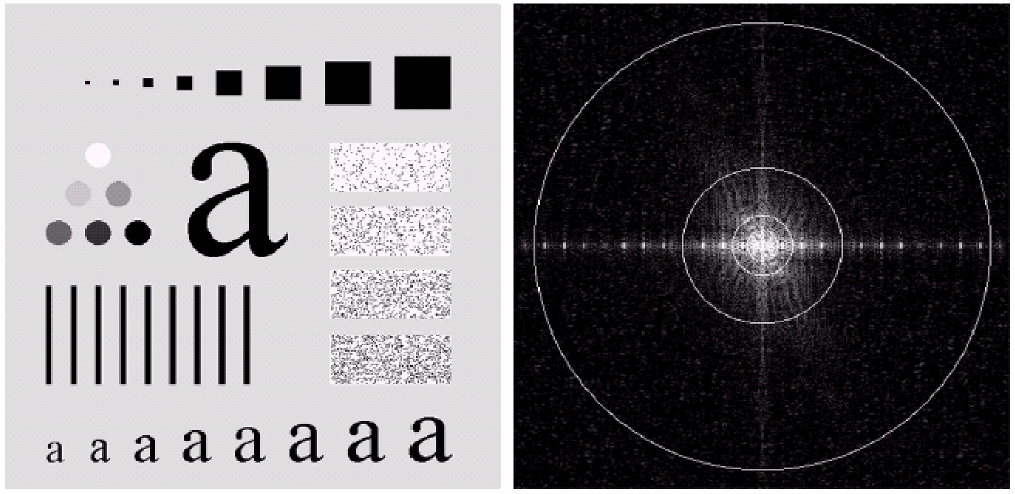
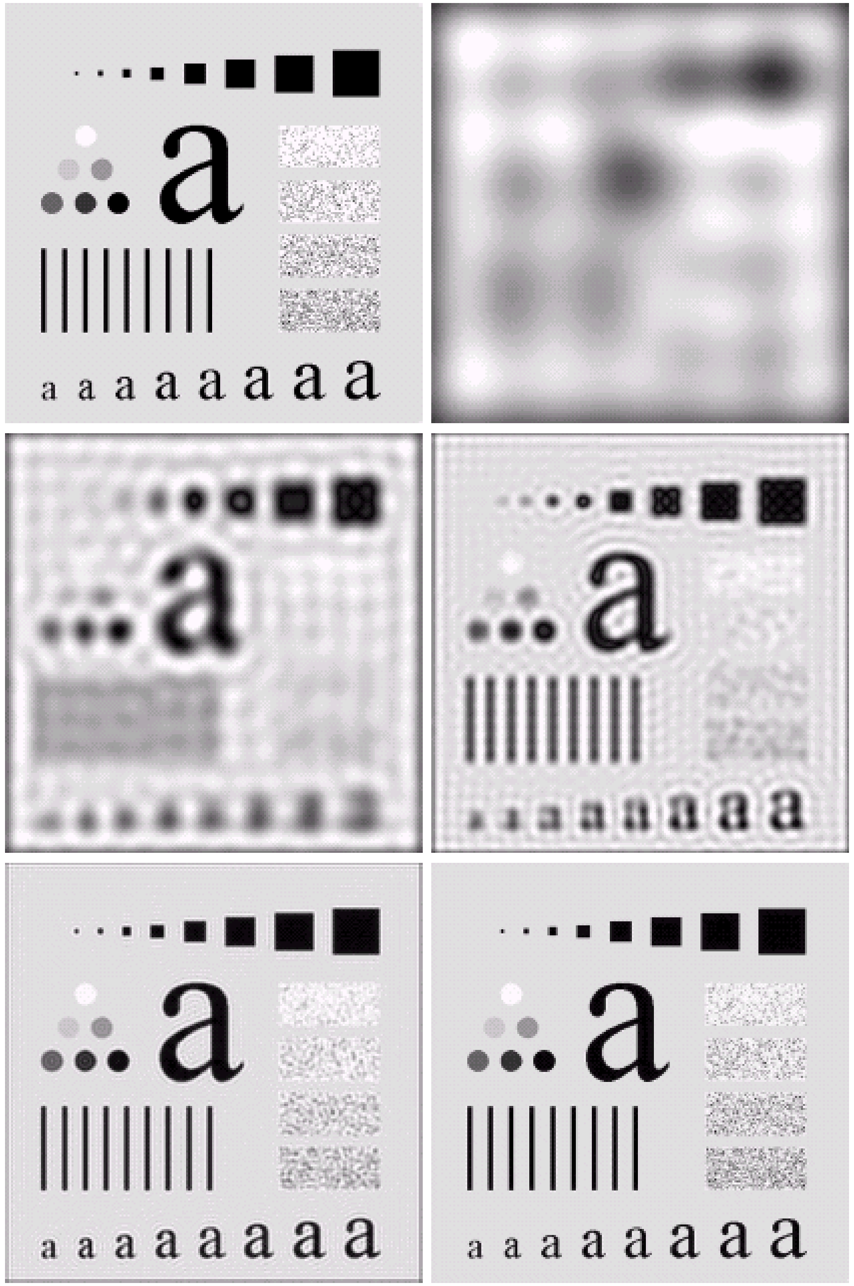


FIGURE 4.11 (a) An image of size 500 × 500 pixels and (b) its Fourier spectrum. The superimposed circles have radii values of 5, 15, 30, 80, and 230, which enclose 92.0, 94.6, 96.4, 98.0, and 99.5% of the image power, respectively.



(a) Original image, (b)-(f) Results of ideal lowpass filtering with cuttoff frequencies set radii of 5, 15, 30, 80, and 230

- Bagaimana cara mengatasi efek *ringing*? Gunakan GLPF atau BLPF.

- Respons frekuensi pada GLPF dan BLPF tidak memiliki transisi yang tajam
  - lebih sesuai untuk penghalusan gambar
  - tidak menimbulkan efek *ringing*.

Gaussian Low Pass Filter (GLPF)

Parameter D0 mengukur penyebaran kurva Gaussian.
Semakin besar nilai D0, semakin besar frekuensi cuttoff.

$$H(u,v) = e^{-D^2(u,v)/2D_0^2}$$

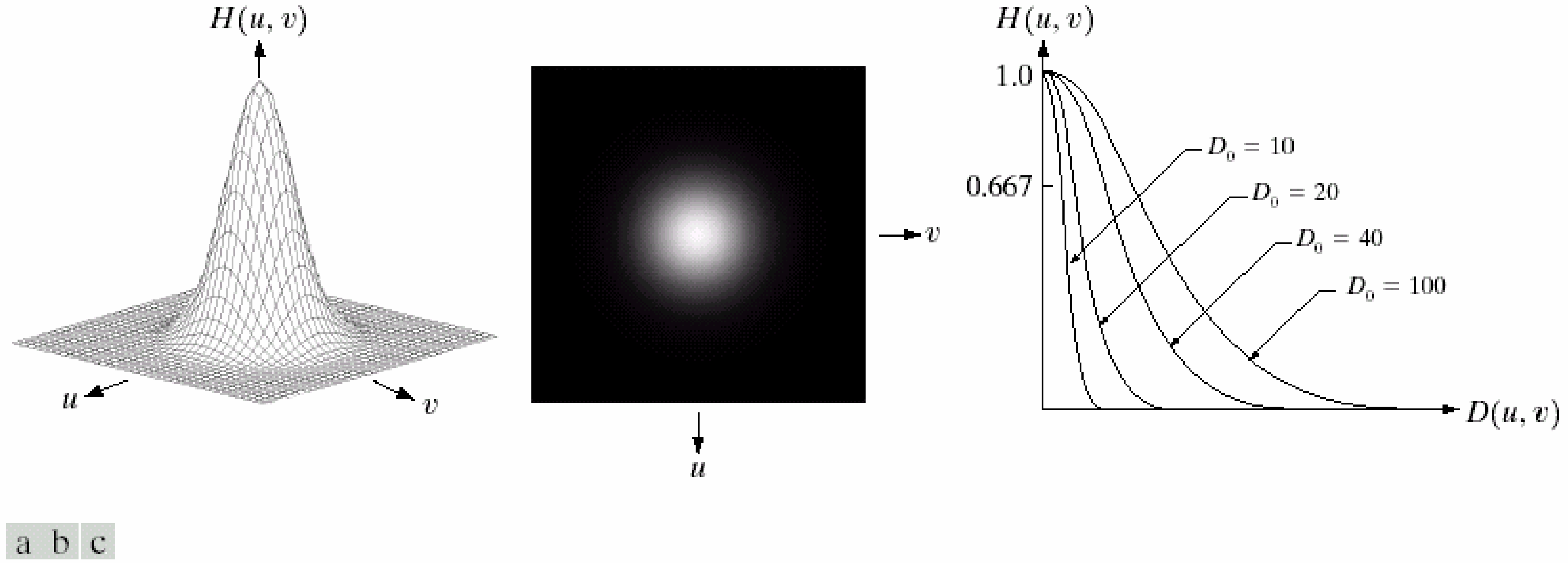


FIGURE 4.17 (a) Perspective plot of a GLPF transfer function. (b) Filter displayed as an image. (c) Filter radial cross sections for various values of D_0.

Butterworth Pass Filter (BLPF)

$$H(u,v) = \frac{1}{1+\left[D(u,v)/D_o\right]^{2n}}$$

n : filter order
D_0 : cutoff frequency

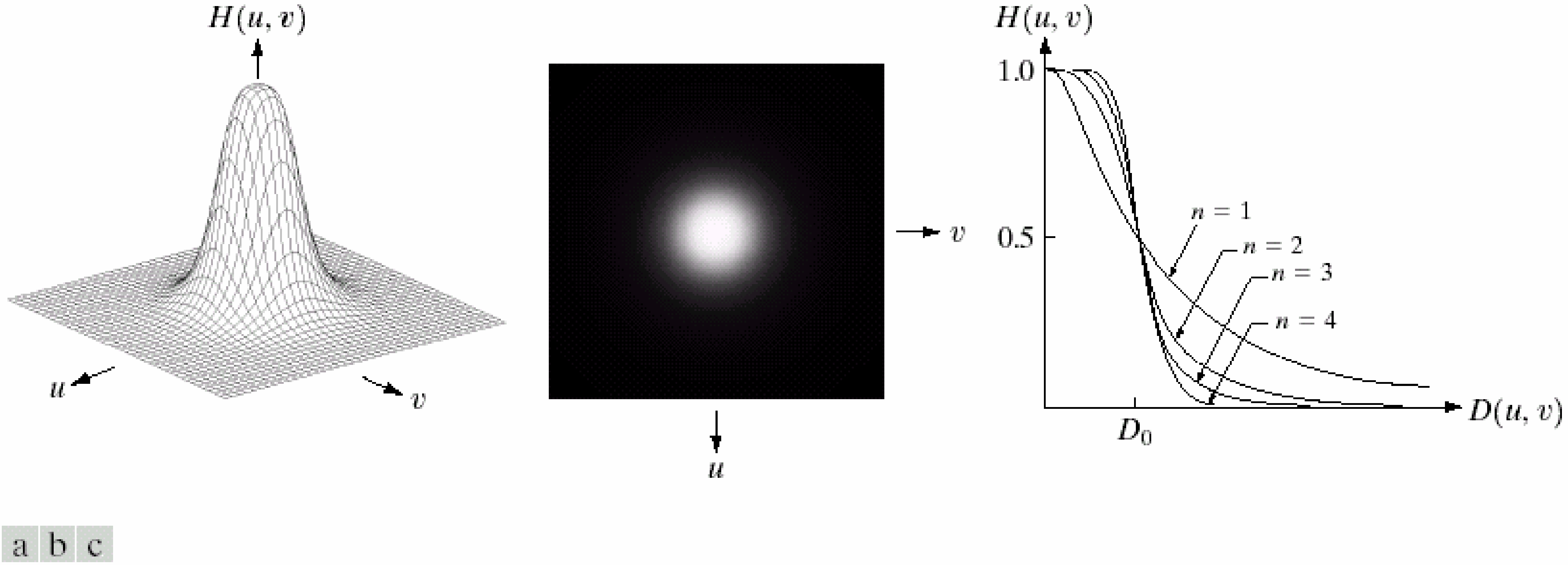


FIGURE 4.14 (a) Perspective plot of a Butterworth lowpass filter transfer function. (b) Filter displayed as an image. (c) Filter radial cross sections of orders 1 through 4.

Catatan:

- Untuk GLPF, maka kode di dalam program ILPF:

```
H = double(D <=D0);
```

diganti menjadi

```
H = exp(-(D.^2)./(2*(D0^2)));
```

- Untuk BLPF orde n, maka kodenya:

```
n = 1;   % default
H = 1./(1 + (D./D0).^(2*n));
```

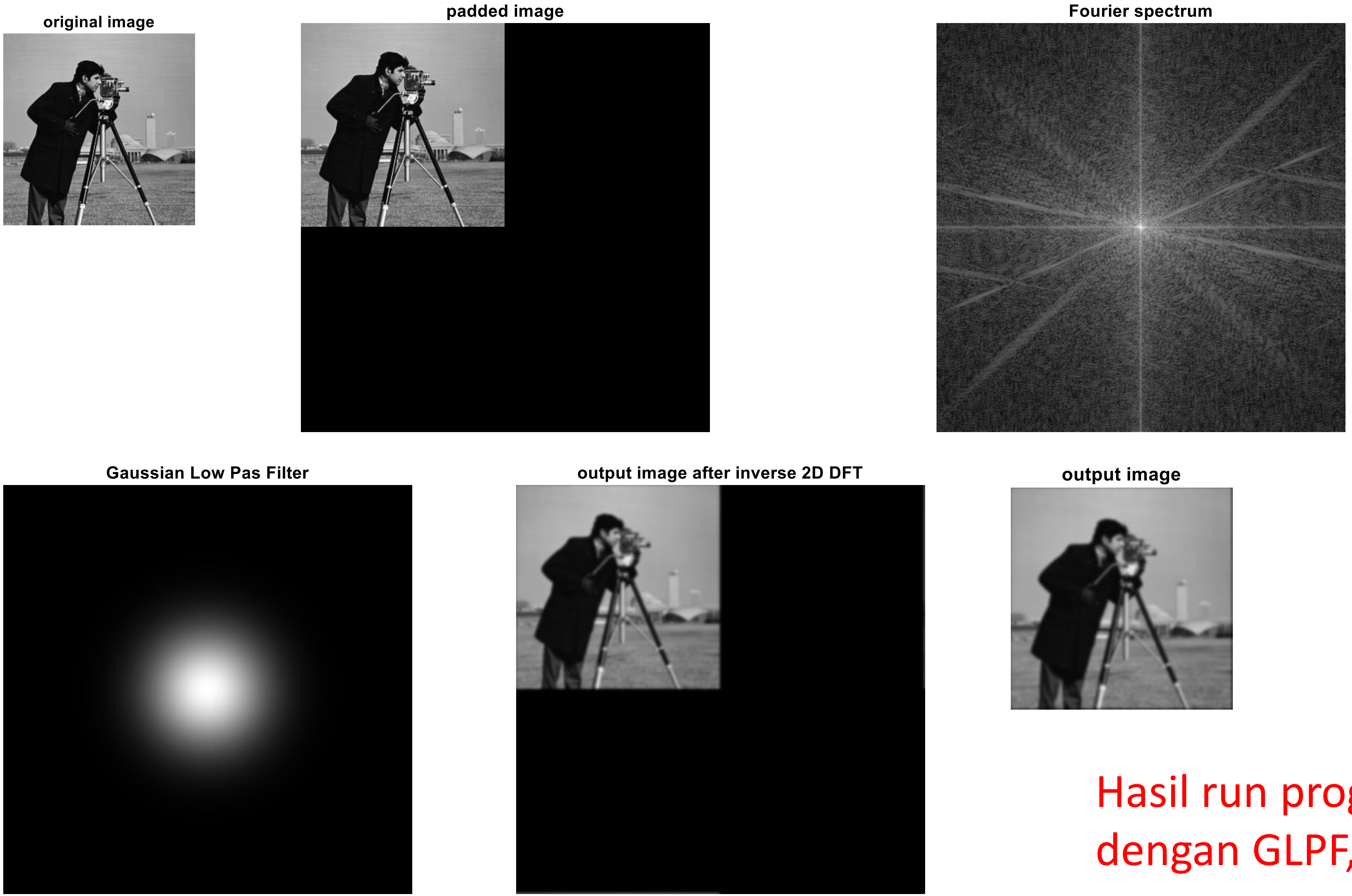
```
% Penapisan dalam ranah frekuensi dengan Gaussian Lowpass Filter (GLPF)
f = imread('cameraman.bmp');
imshow(f);
[M,N] = size(f);
%Step 1: Tentukan parameter padding, biasanya untuk citra f(x,y)
%        berukuran M x N, parameter padding P dan Q biasanya P = 2M and Q = 2N.
P = 2*M;
Q = 2*N;

%Step 2: Bentuklah citra padding fp(x,y) berukuran  P X Q dengan
% menambahkan pixel-pixel bernilai nol pada f(x, y).
f = im2double(f);
for i = 1:P
    for j = 1:Q
        if i <= M && j<= N
            fp(i,j) = f(i,j);
        else
            fp(i,j) = 0;
        end
    end
end
imshow(f);title('original image');
figure; imshow(fp);title('padded image');
```

```
%Step 3: Lakukan transformasi Fourier pada fp(x, y) dan tampilkan Fourier
Spectrum
F = fftshift(fft2(fp)); % move the origin of the transform to the center of
the frequency rectangle
S2 = log(1+abs(F)); % use abs to compute the magnitude (handling imaginary)
and use log to brighten display
figure, imshow(S2,[]); title('Fourier spectrum');

%Step 4: Bangkitkan fungsi penapis H berukuran P x Q, misalkan penapis
%yang digunakan adalah Ideal Lowpass Filter (ILPF)

D0 = 50;    % cut-off frequency
% Set up range of variables.
u = 0:(P-1);
v = 0:(Q-1);

% Compute the indices for use in meshgrid
idx = find(u > P/2);
u(idx) = u(idx) - P;
idy = find(v > Q/2);
v(idy) = v(idy) - Q;
```

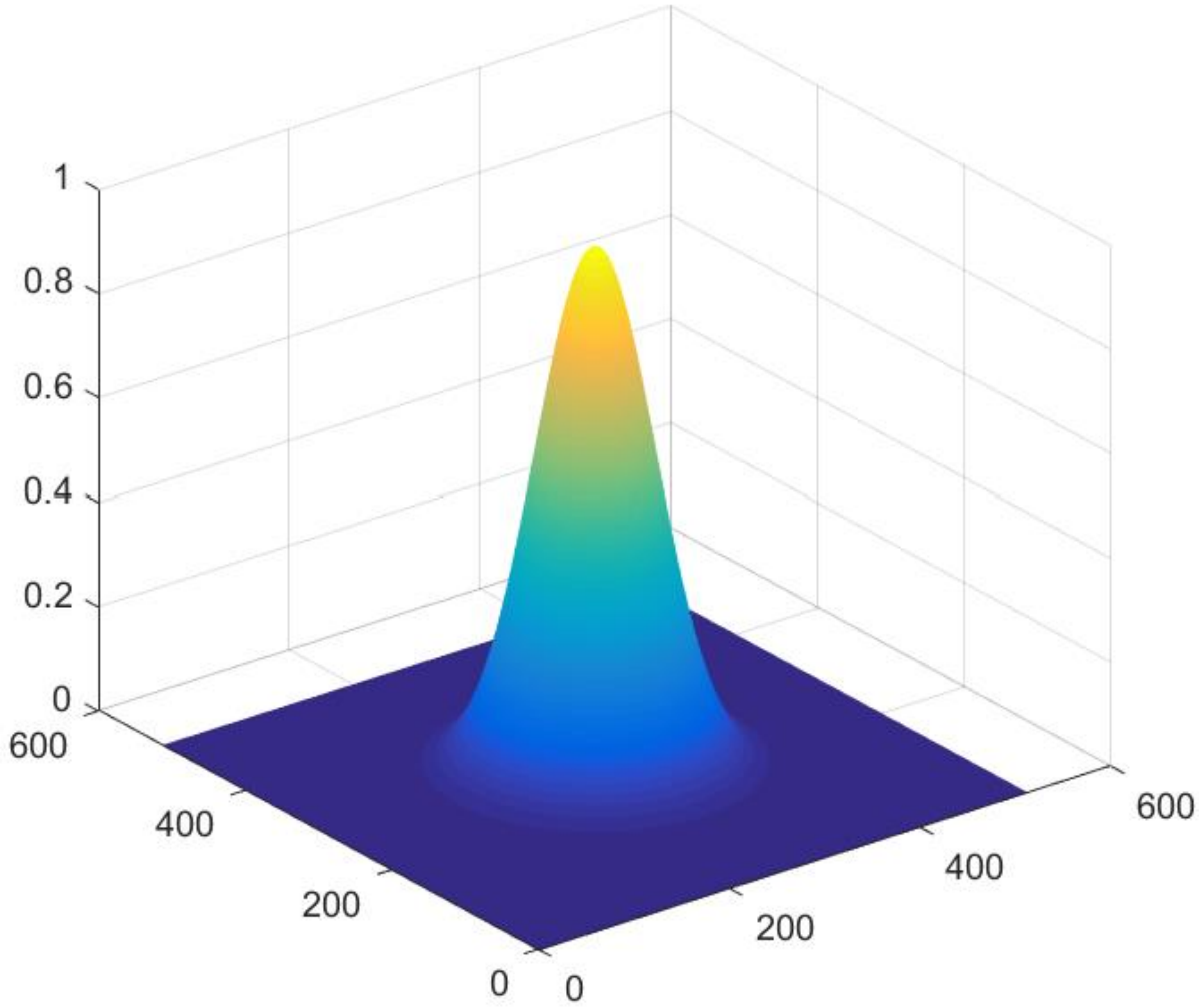
```
% Compute the meshgrid arrays
[V, U] = meshgrid(v, u);
D = sqrt(U.^2 + V.^2);

H = exp(-(D.^2)./(2*(D0^2)));
H = fftshift(H); figure;imshow(H);title('LPF Gaussian Mask');
figure, mesh(H);

%Step 5: Kalikan F dengan H
G = H.*F;
G1 = ifftshift(G);

%Step 6: Ambil bagian real dari inverse FFT of G:
G2 = real(ifft2(G1)); % apply the inverse, discrete Fourier transform
figure; imshow(G2);title('output image after inverse 2D DFT');

%Step 7: Potong bagian kiri atas sehingga menjadi berukuran citra semula
G2 = G2(1:M, 1:N); % Resize the image to undo padding
figure, imshow(G2); title('output image');
```

Sumber: Program di atas merupakan modifikasi dari program yang diambil dari sini:

1. https://www.cs.uregina.ca/Links/class-info/425-nova/Lab5/index.html
2. https://www.mathworks.com/matlabcentral/fileexchange/53250-filtering-of-an-image-in-frequency-domain



Hasil run program dengan GLPF, D0 = 50



mesh(H)

```
% Penapisan dalam ranah frekuensi dengan Butterworth Lowpass Filter (BLPF)
f = imread('cameraman.bmp');
imshow(f);
[M,N] = size(f);
%Step 1: Tentukan parameter padding, biasanya untuk citra f(x,y)
%        berukuran M x N, parameter padding P dan Q biasanya P = 2M and Q = 2N.
P = 2*M;
Q = 2*N;

%Step 2: Bentuklah citra padding fp(x,y) berukuran  P X Q dengan
% menambahkan pixel-pixel bernilai nol pada f(x, y).
f = im2double(f);
for i = 1:P
    for j = 1:Q
        if i <= M && j<= N
            fp(i,j) = f(i,j);
        else
            fp(i,j) = 0;
        end
    end
end
imshow(f);title('original image');
figure; imshow(fp);title('padded image');
```

```
%Step 3: Lakukan transformasi Fourier pada fp(x, y) dan tampilkan Fourier
Spectrum
F = fftshift(fft2(fp)); % move the origin of the transform to the center of
the frequency rectangle
S2 = log(1+abs(F)); % use abs to compute the magnitude (handling imaginary)
and use log to brighten display
figure, imshow(S2,[]); title('Fourier spectrum');

%Step 4: Bangkitkan fungsi penapis H berukuran P x Q, misalkan penapis
%yang digunakan adalah Ideal Lowpass Filter (ILPF)

D0 = 50;    % cut-off frequency
% Set up range of variables.
u = 0:(P-1);
v = 0:(Q-1);

% Compute the indices for use in meshgrid
idx = find(u > P/2);
u(idx) = u(idx) - P;
idy = find(v > Q/2);
v(idy) = v(idy) - Q;
```

```
% Compute the meshgrid arrays
[V, U] = meshgrid(v, u);
D = sqrt(U.^2 + V.^2);

n = 2;    % default
H = 1./(1 + (D./D0).^(2*n));
H = fftshift(H); figure;imshow(H);title('LPF Butterworth Mask');
figure, mesh(H);

%Step 5: Kalikan F dengan H
G = H.*F;
G1 = ifftshift(G);

%Step 6: Ambil bagian real dari inverse FFT of G:
G2 = real(ifft2(G1)); % apply the inverse, discrete Fourier transform
figure; imshow(G2);title('output image after inverse 2D DFT');

%Step 7: Potong bagian kiri atas sehingga menjadi berukuran citra semula
G2 = G2(1:M, 1:N); % Resize the image to undo padding
figure, imshow(G2); title('output image');
```

Sumber: Program di atas merupakan modifikasi dari program yang diambil dari sini:

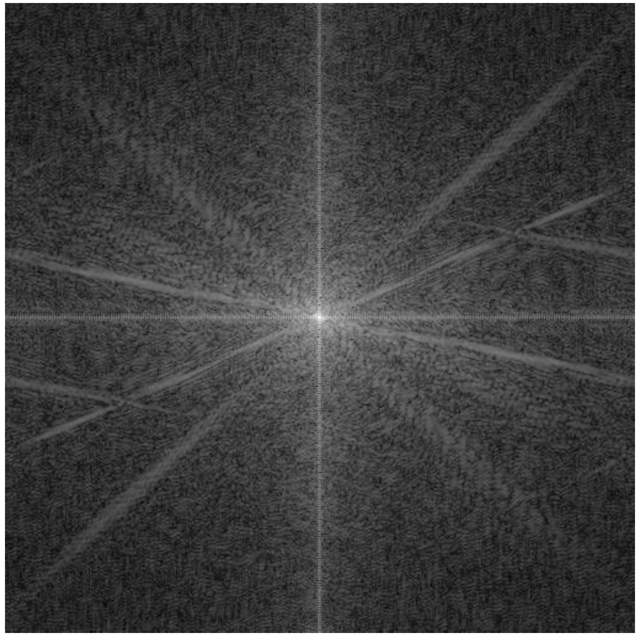
1. https://www.cs.uregina.ca/Links/class-info/425-nova/Lab5/index.html
2. https://www.mathworks.com/matlabcentral/fileexchange/53250-filtering-of-an-image-in-frequency-domain

```
% Compute the meshgrid arrays
[V, U] = meshgrid(v, u);
D = sqrt(U.^2 + V.^2);

n = 2;    % default
H = 1./(1 + (D./D0).^(2*n));
H = fftshift(H); figure;imshow(H);title('Butterworth Low Pass Filter');
figure, mesh(H);

%Step 5: Kalikan F dengan H
H = ifftshift(H);
LPF_f = H.*F;

%Step 6: Ambil bagian real dari inverse FFT of G:
LPF_f2=real(ifft2(LPF_f)); % apply the inverse, discrete Fourier transform
figure; imshow(LPF_f2);title('output image after inverse 2D DFT');

%Step 7: Potong bagian kiri atas sehingga menjadi berukuran citra semula
LPF_f2=LPF_f2(1:M, 1:N); % Resize the image to undo padding
figure, imshow(LPF_f2); title('output image');
```

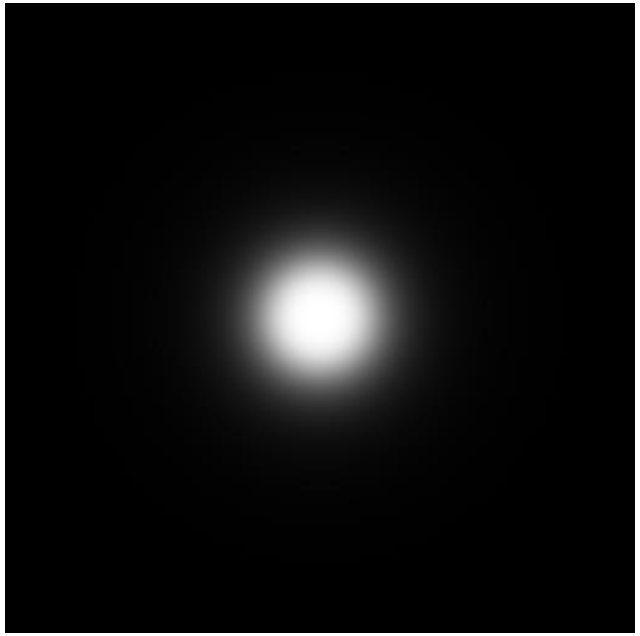
original image

padded image

Fourier spectrum



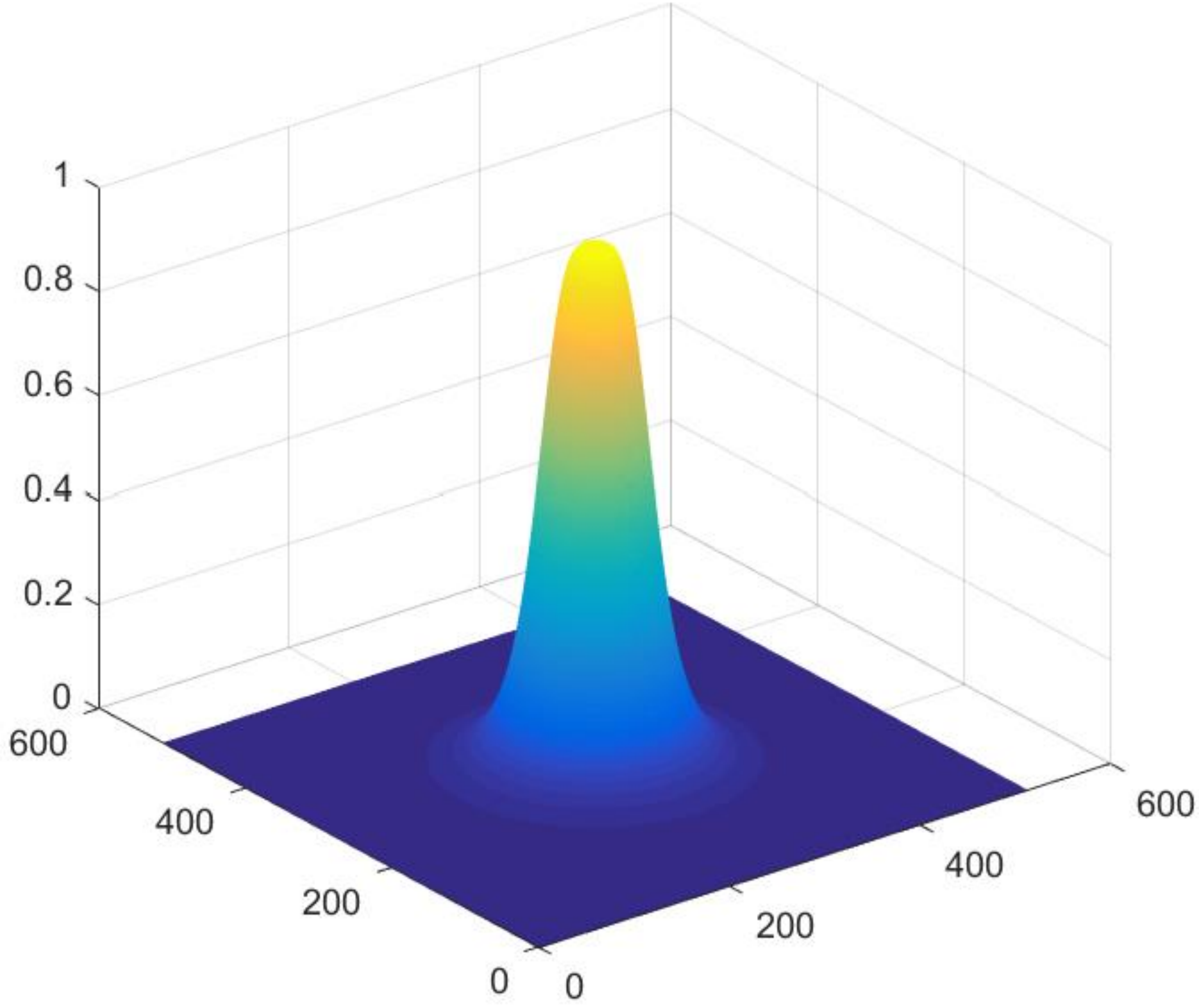
Butterworth Low Pas Filter



output image after inverse 2D DFT

output image

Hasil run program dengan BLPF, n = 2, D0 = 50



mesh(H)

- Perbandingan hasil

output image

ILPF

output image

GLPF

output image

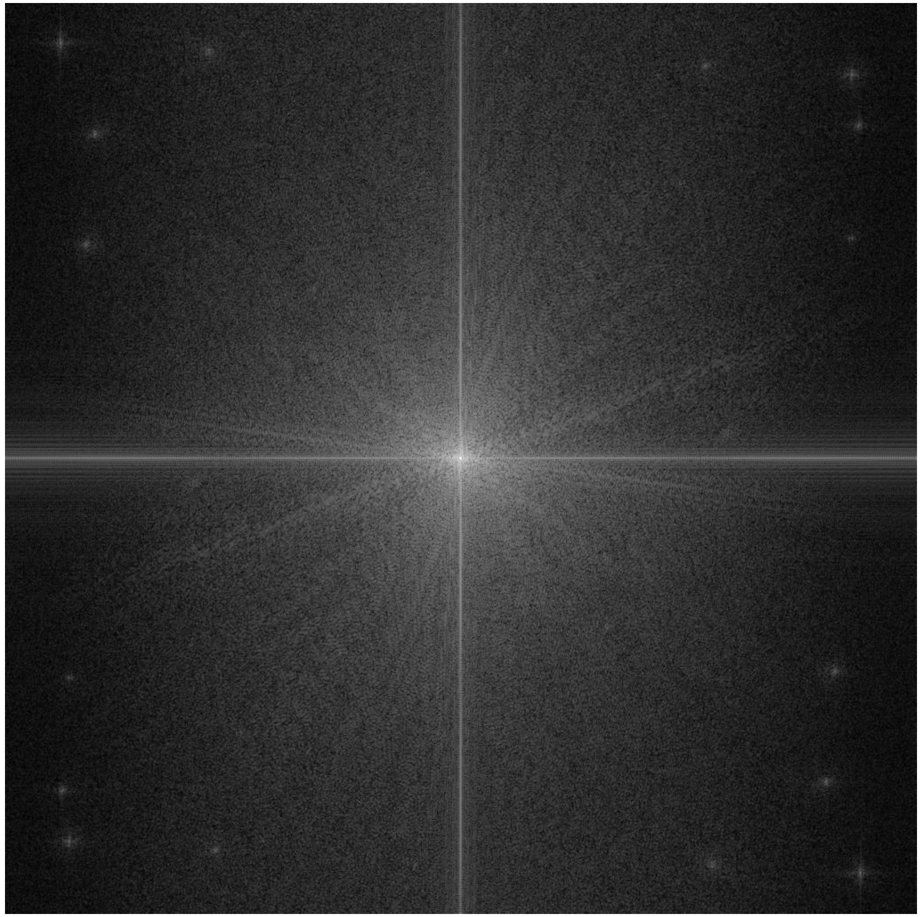
BLPF

Contoh hasil run pada citra *pepper* dengan Gaussian Low Pass Filter:

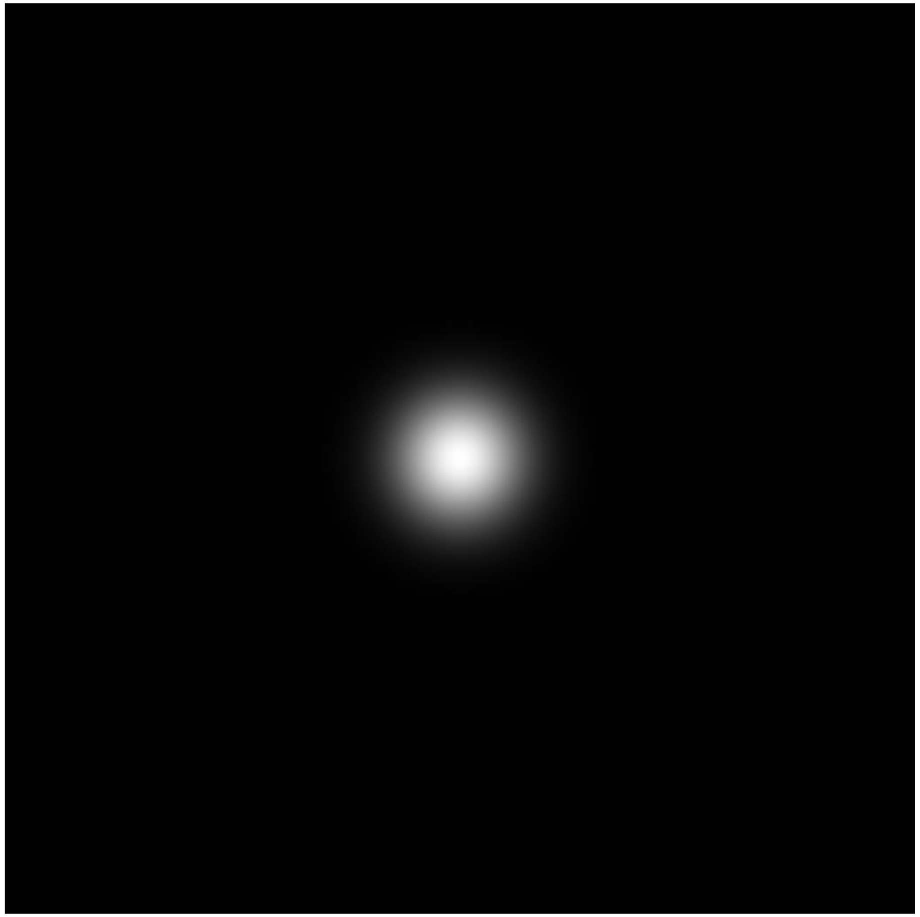
original image

padded image

Fourier spectrum



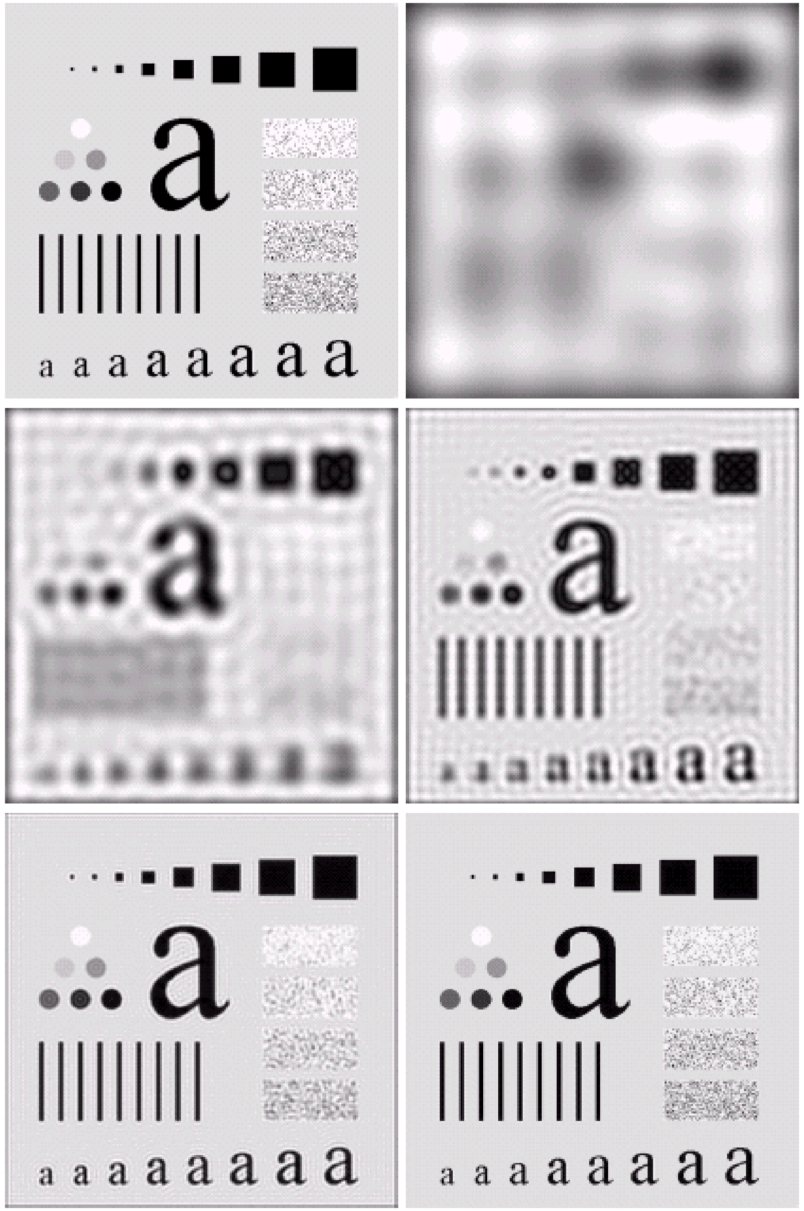
Gaussian Low Pas Filter



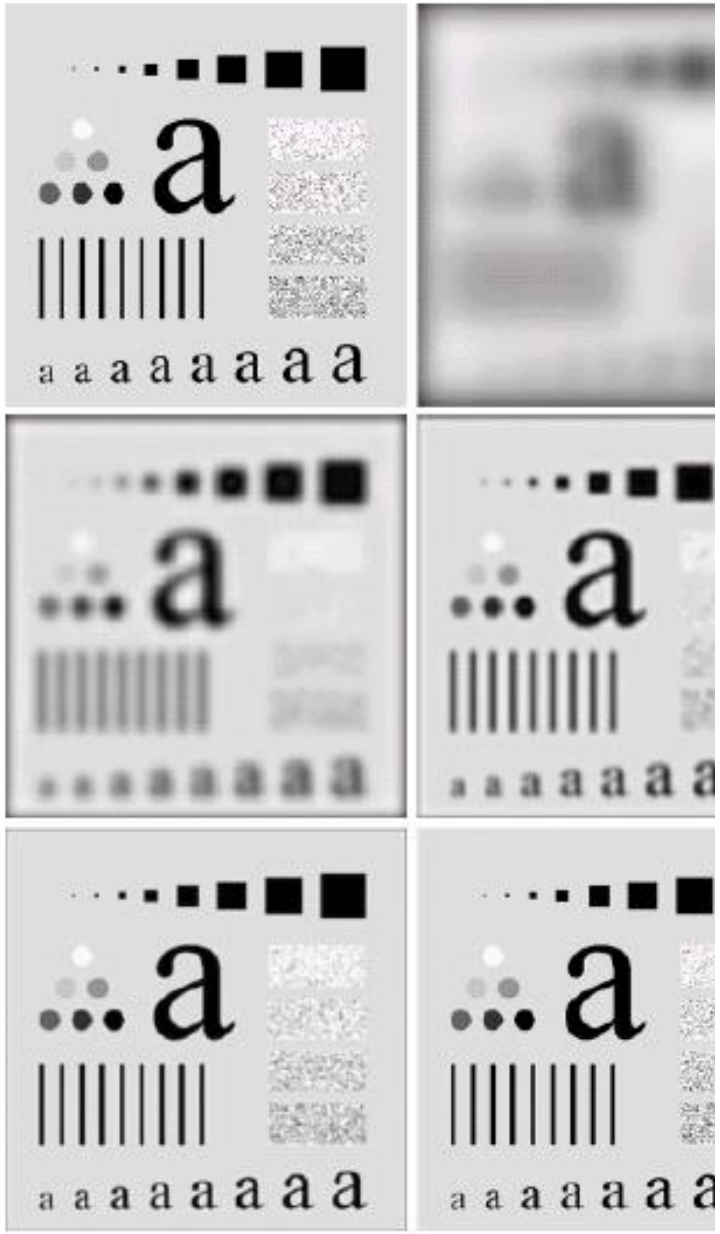
output image after inverse 2D DFT

output image

Contoh hasil ILPF



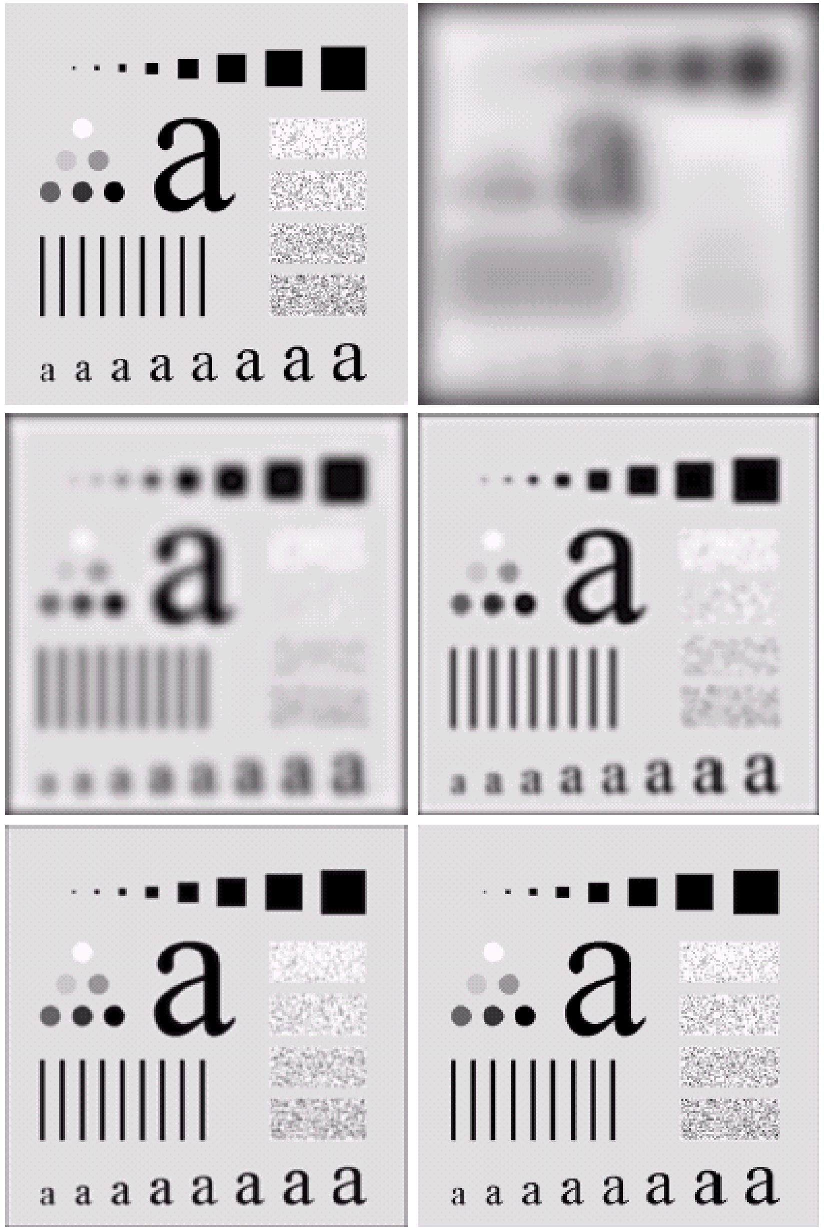
Contoh hasil GLPF



a b
c d
e f

FIGURE 4.15 (a) Original image. (b)–(f) Results of filtering with BLPFs
with cutoff frequencies at radii of 5, 15, 30, 80, and 230, as shown in F
Compare with Fig. 4.12.

Contoh hasil BLPF



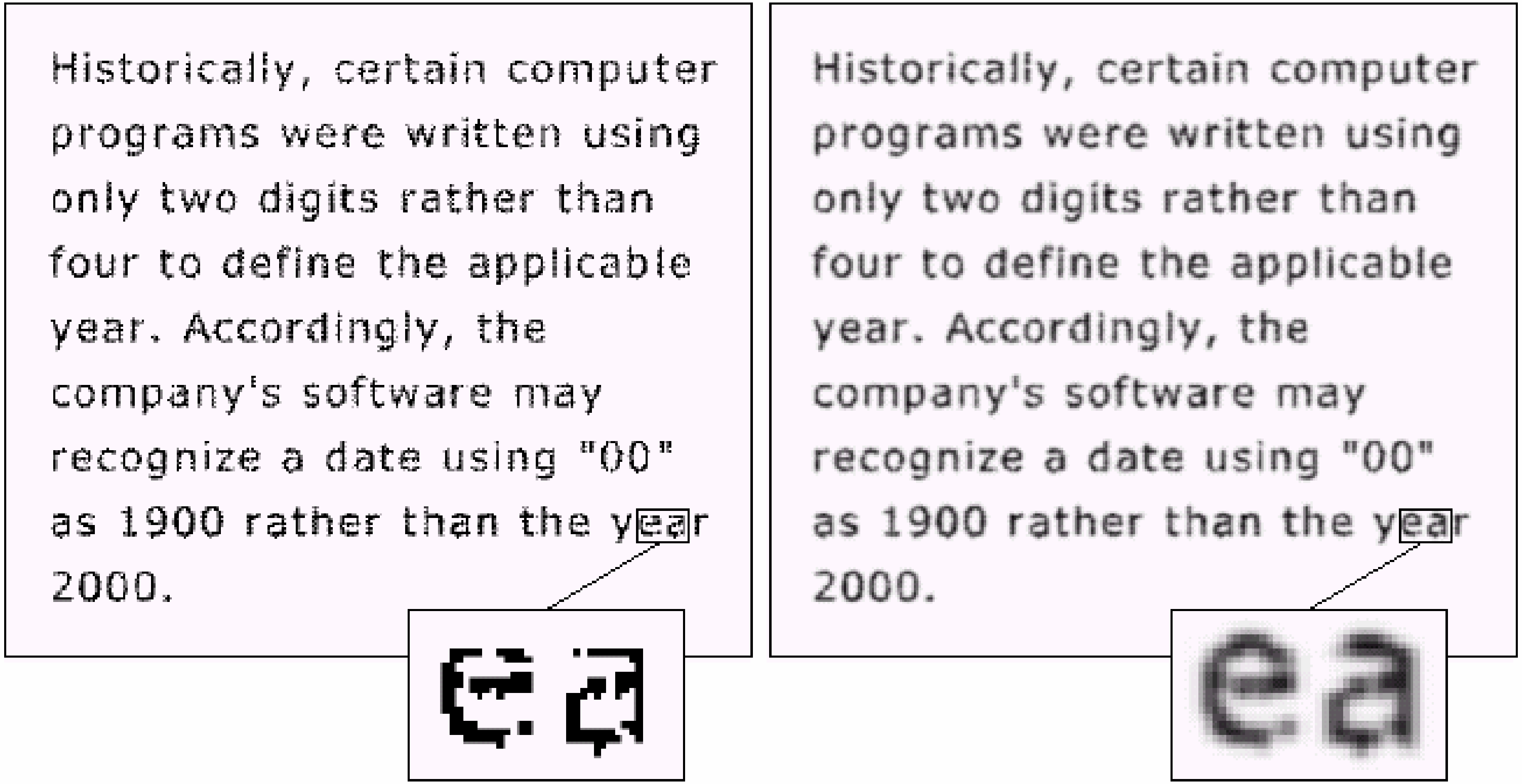
a b
c d
e f

FIGURE 4.15 (a) Original image. (b)–(f) Results of filtering with BLPFs of order 2, with cutoff frequencies at radii of 5, 15, 30, 80, and 230, as shown in Fig. 4.11(b). Compare with Fig. 4.12.

GLPF dapat menyambung bagian-bagian yang putus pada citra font karakter:

a b

FIGURE 4.19 (a) Sample text of poor resolution (note broken characters in magnified view). (b) Result of filtering with a GLPF (broken character segments were joined).



a b c

FIGURE 4.20 (a) Original image (1028 × 732 pixels). (b) Result of filtering with a GLPF with $D_0 = 100$. (c) Result of filtering with a GLPF with $D_0 = 80$. Note reduction in skin fine lines in the magnified sections of (b) and (c).

a b c

FIGURE 4.21 (a) Image showing prominent scan lines. (b) Result of using a GLPF with $D_0 = 30$. (c) Result of using a GLPF with $D_0 = 10$. (Original image courtesy of NOAA.)

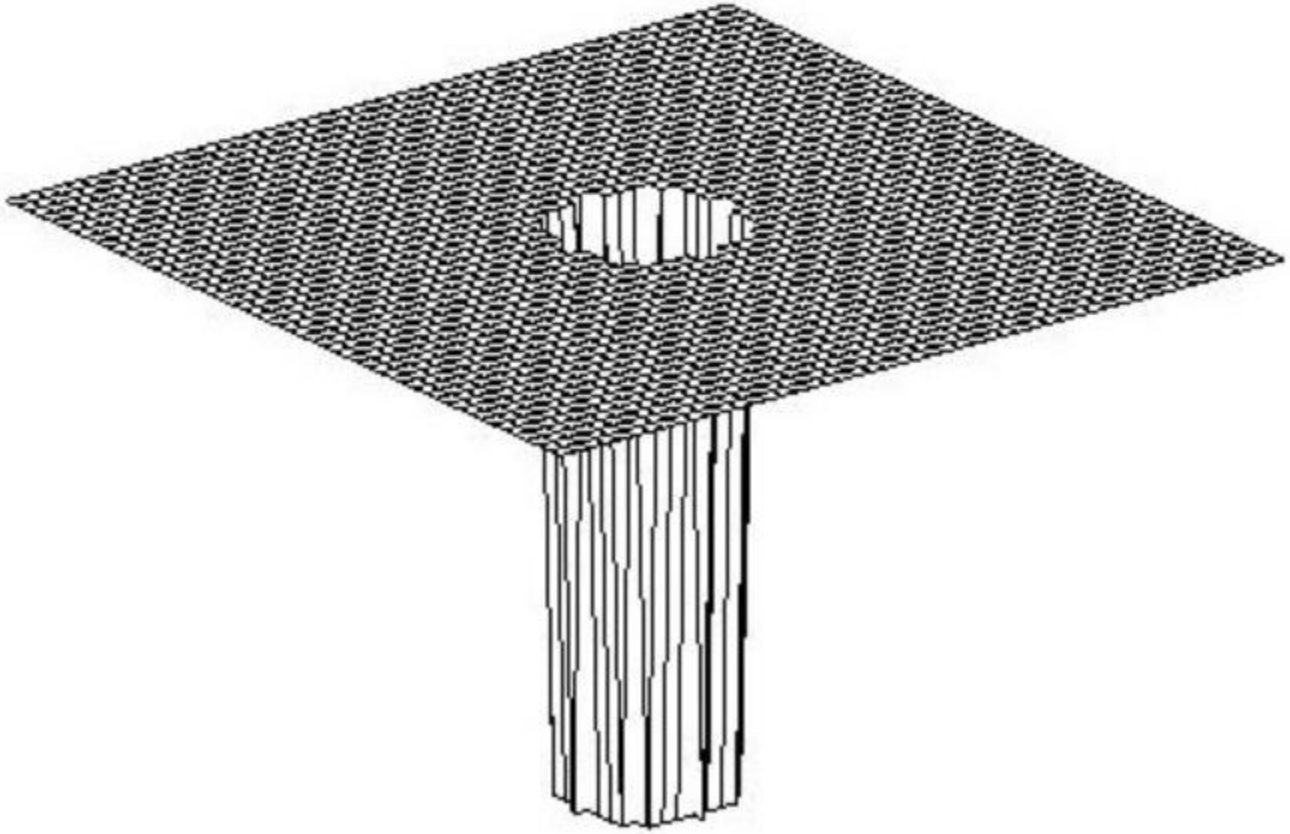
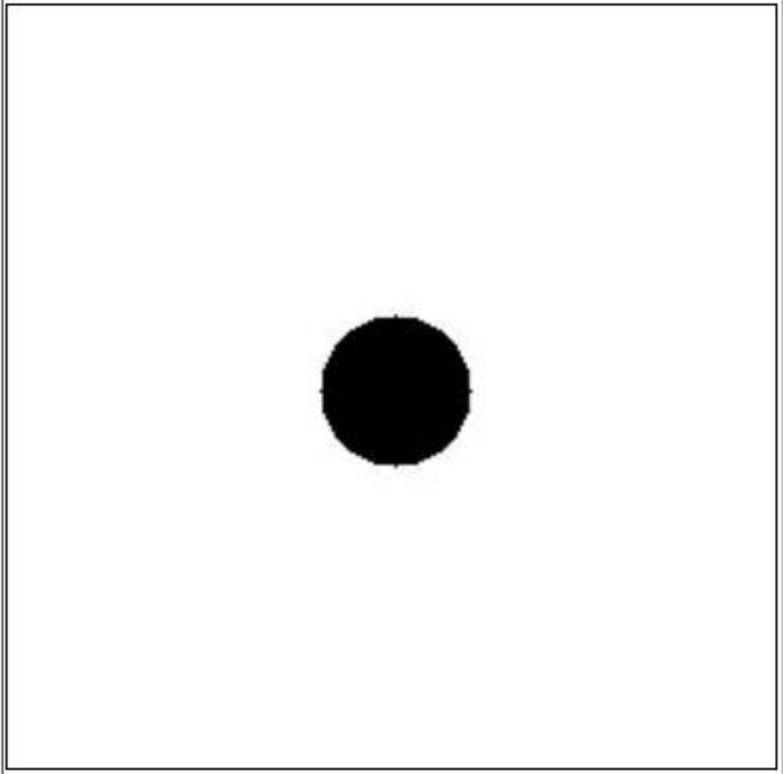
Highpass filter (HPF) dalam ranah frekuensi

- Penapis lolos tinggi (*highpass filter*) bertujuan menekan komponen berfrekuensi rendah dan dan meloloskan (sekaligus memperkuat) komponen berfrekuensi tinggi.

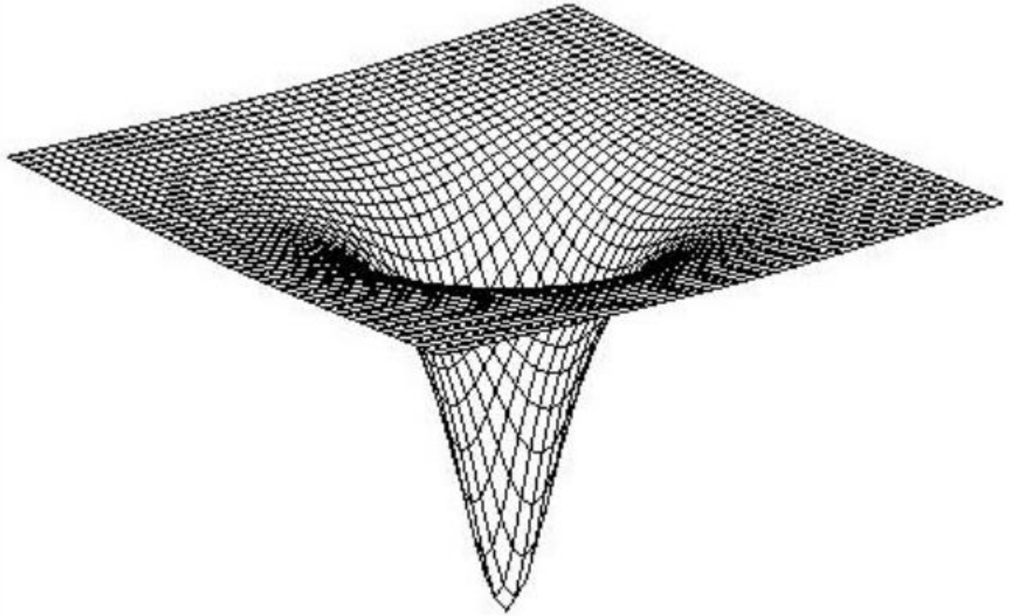
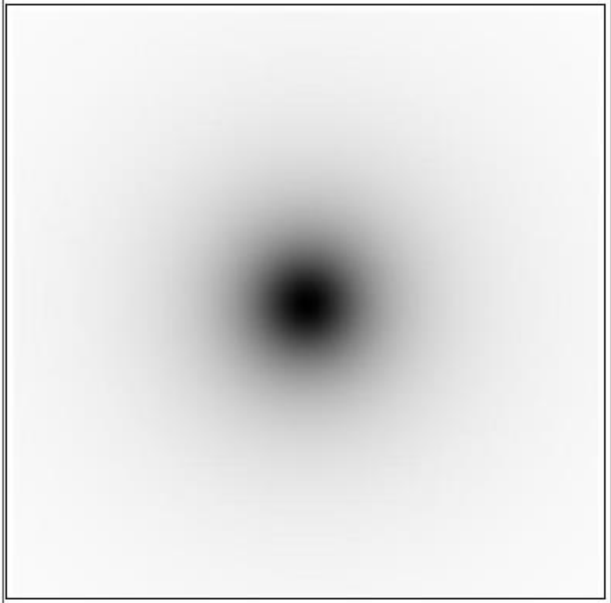
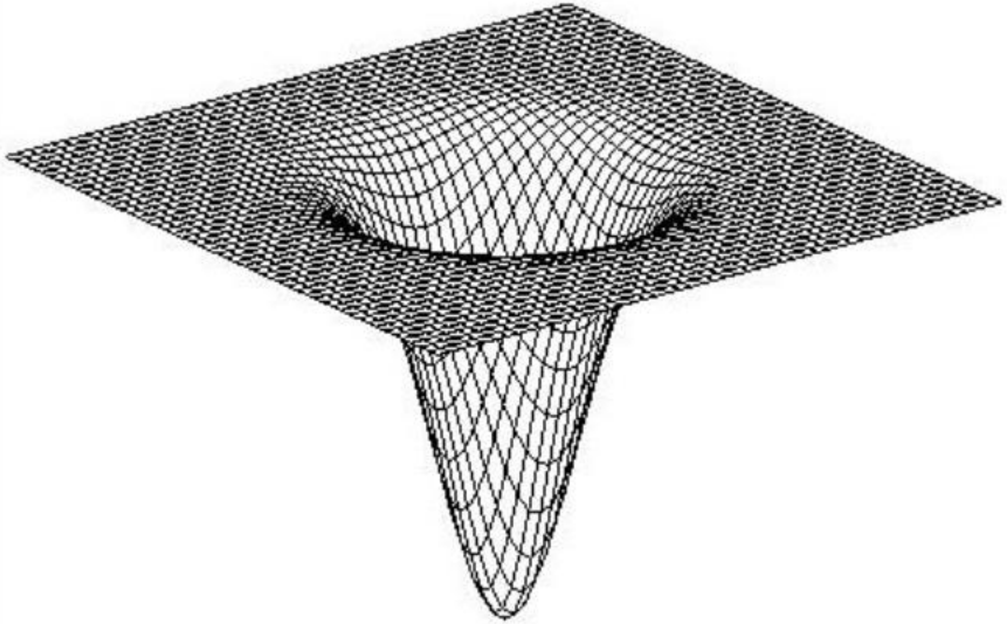
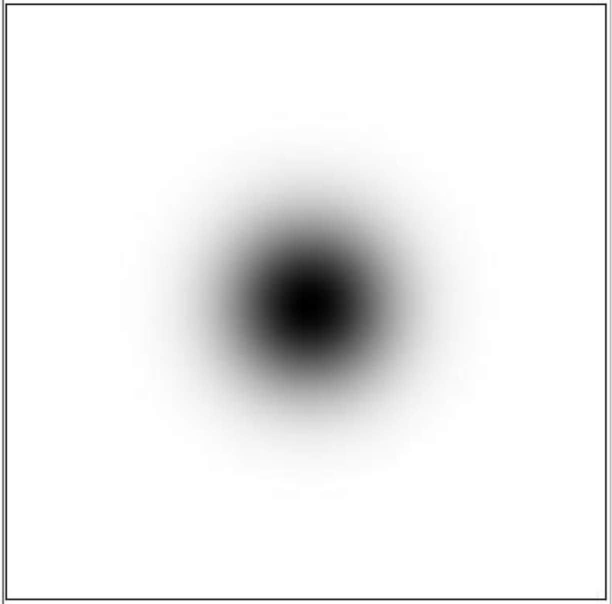
- Menghasilkan efek penajaman pada tepi (edge) citra (atau *sharpened image*)
- Hubungan antara penapis lolos tinggi (H_{hp}) dan penapis lolos rendah (H_{lp})

$$H_{hp}(u,v) = 1 - H_{lp}(u,v)$$

- Tiga buah HPF yang utama:
  1. *Ideal highpass filter* (IHPF)
  2. *Butterworth highpass filter* (BHPF)
  3. *Gaussian highpass filter* (GHPF)

Highpass Filter	Mesh	Image
Ideal		

$$H(u,v)=\begin{cases}0 & \text{if } D(u,v)\le D_0\\ 1 & \text{if } D(u,v)>D_0\end{cases}$$

Butterworth		
Gaussian		

$$H(u,v) = \frac{1}{1 + \left[D_0 / D(u,v)\right]^{2n}}$$

$$H(u,v) = 1 - e^{-D^2(u,v)/2D_0^2}$$

Ideal highpass filter

$$H(u,v)=\begin{cases}0 \text{ if } D(u,v)\le D_0\\ 1 \text{ if } D(u,v)>D_0\end{cases}$$

Butterworth highpass filter

$$H(u,v)=\frac{1}{1+\left[D_0/D(u,v)\right]^{2n}}$$

Gaussian highpass filter

$$H(u,v)=1-e^{-D^2(u,v)/2D_0^2}$$

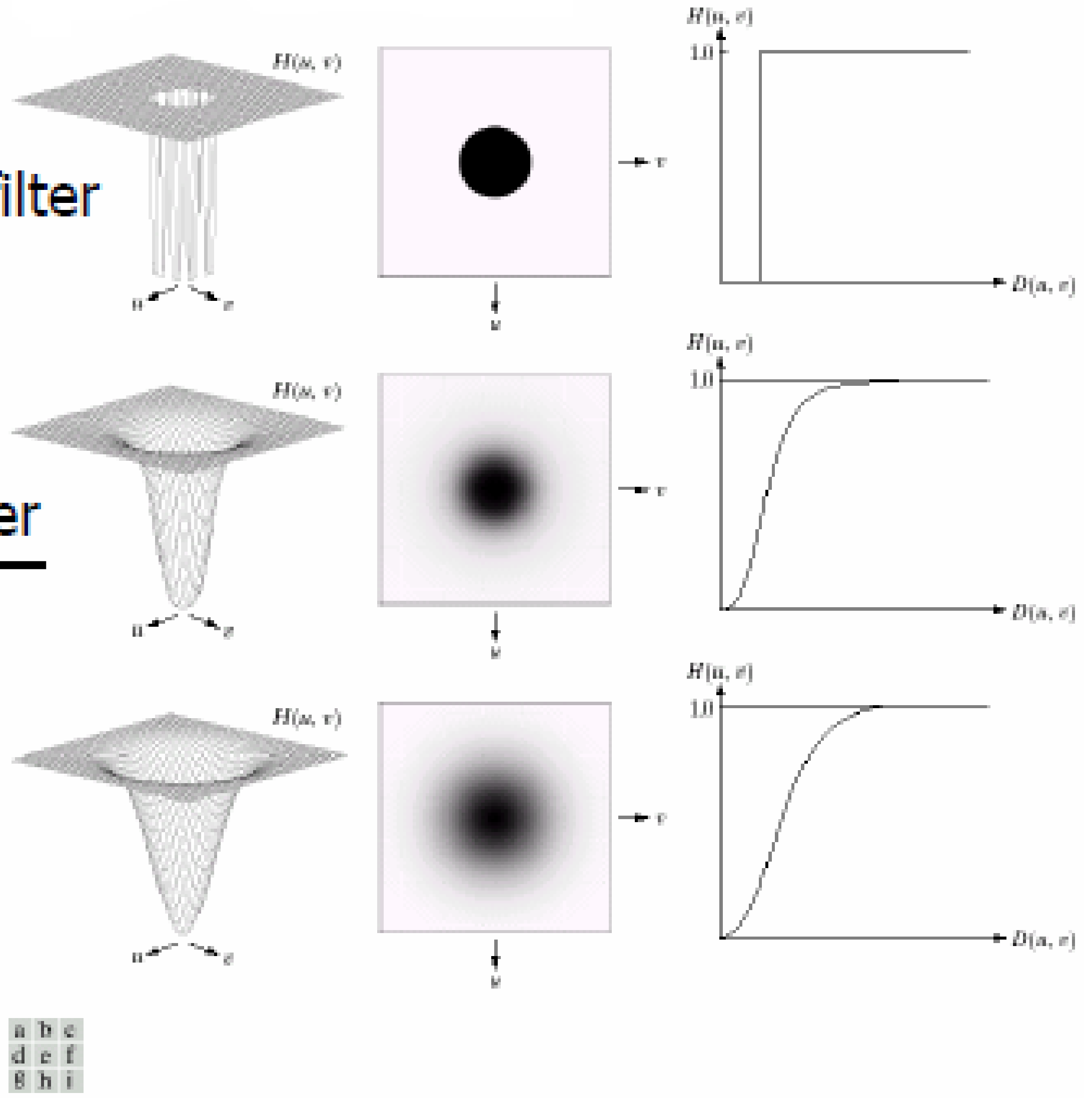
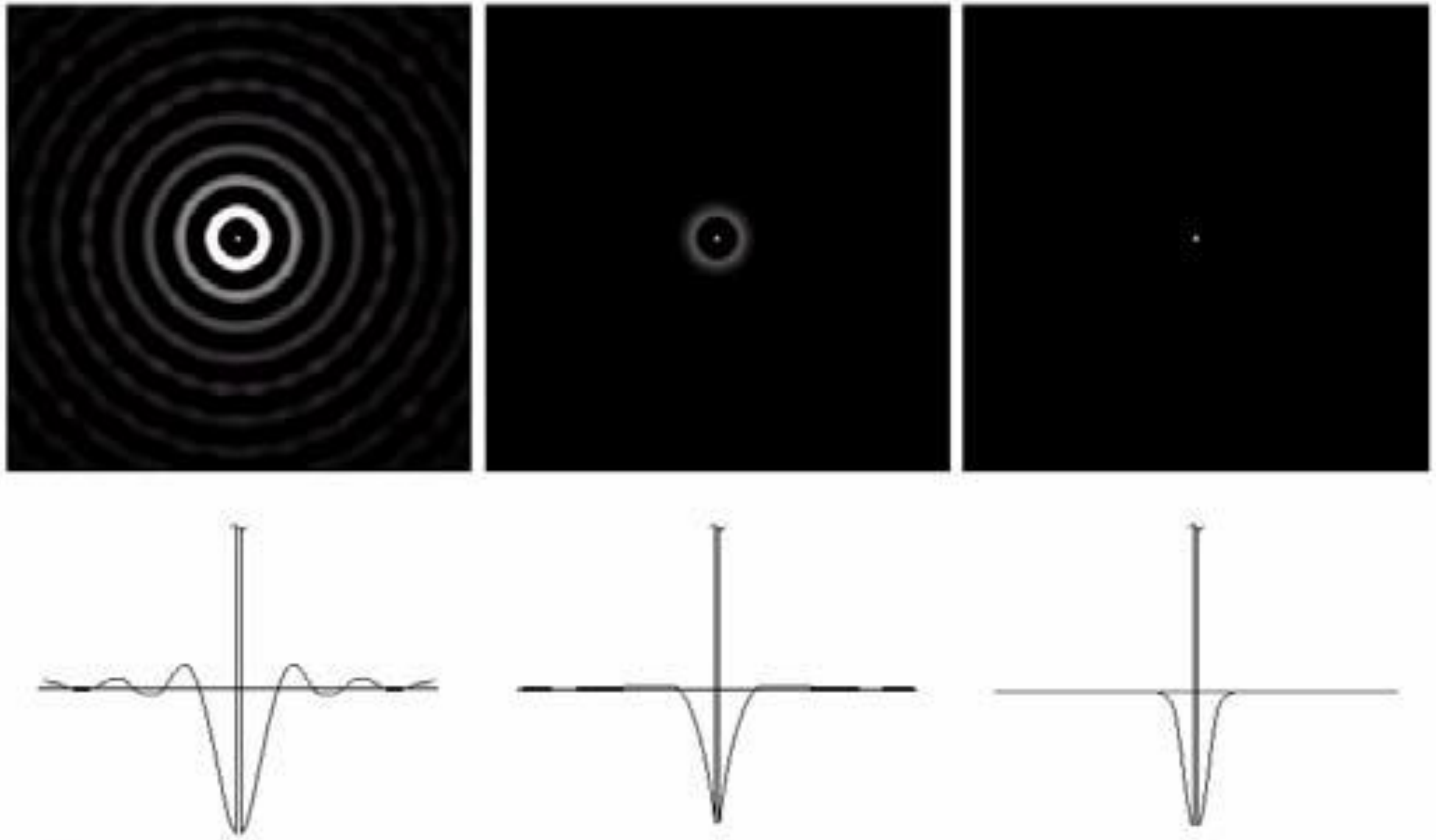


FIGURE 4.22 Top row: Perspective plot, image representation, and cross section of a typical ideal highpass filter. Middle and bottom rows: The same sequence for typical Butterworth and Gaussian highpass filters.

Representasi Spasial Ideal, Butterworth dan Gaussian Highpass Filter



a b c

FIGURE 4.23 Spatial representations of typical (a) ideal, (b) Butterworth, and (c) Gaussian frequency domain highpass filters, and corresponding gray-level profiles.

```
% Penapisan dalam ranah frekuensi dengan Gaussian Highpass Filter (GHPF)
f = imread('cameraman.bmp');
imshow(f);
[M,N] = size(f);
%Step 1: Tentukan parameter padding, biasanya untuk citra f(x,y)
%         berukuran M x N, parameter padding P dan Q biasanya P = 2M and Q = 2N.
P = 2*M;
Q = 2*N;

%Step 2: Bentuklah citra padding fp(x,y) berukuran  P X Q dengan
% menambahkan pixel-pixel bernilai nol pada f(x, y).
f = im2double(f);
for i = 1:P
    for j = 1:Q
        if i <= M && j<= N
            fp(i,j) = f(i,j);
        else
            fp(i,j) = 0;
        end
    end
end
imshow(f);title('original image');
figure; imshow(fp);title('padded image');
```

```
%Step 3: Lakukan transformasi Fourier pada fp(x, y) dan tampilkan Fourier
Spectrum
F = fftshift(fft2(fp)); % move the origin of the transform to the center of
the frequency rectangle
S2 = log(1+abs(F)); % use abs to compute the magnitude (handling imaginary)
and use log to brighten display
figure, imshow(S2,[]); title('Fourier spectrum');

%Step 4: Bangkitkan fungsi penapis H berukuran P x Q, misalkan penapis
%yang digunakan adalah Ideal Lowpass Filter (ILPF)

D0 = 50;    % cut-off frequency
% Set up range of variables.
u = 0:(P-1);
v = 0:(Q-1);

% Compute the indices for use in meshgrid
idx = find(u > P/2);
u(idx) = u(idx) - P;
idy = find(v > Q/2);
v(idy) = v(idy) - Q;
```

```
% Compute the meshgrid arrays
[V, U] = meshgrid(v, u);
D = sqrt(U.^2 + V.^2);

H = exp(-(D.^2)./(2*(D0^2)));
H = 1 - H;
H = fftshift(H); figure;imshow(H);title('LPF Gaussian Mask');
figure, mesh(H);

%Step 5: Kalikan F dengan H
G = H.*F;
G1 = ifftshift(G);

%Step 6: Ambil bagian real dari inverse FFT of G:
G2 = real(ifft2(G1)); % apply the inverse, discrete Fourier transform
figure; imshow(G2);title('output image after inverse 2D DFT');

%Step 7: Potong bagian kiri atas sehingga menjadi berukuran citra semula
G2 = G2(1:M, 1:N); % Resize the image to undo padding
figure, imshow(G2); title('output image');
```

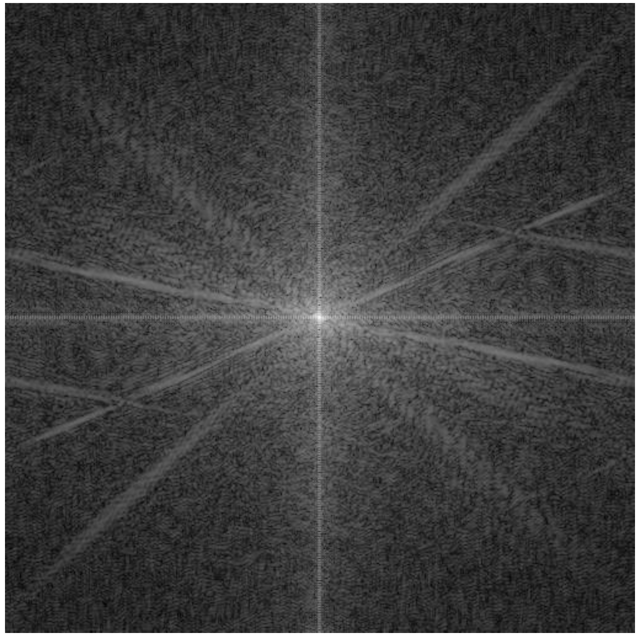
Sumber: Program di atas merupakan modifikasi dari program yang diambil dari sini:

1. https://www.cs.uregina.ca/Links/class-info/425-nova/Lab5/index.html
2. https://www.mathworks.com/matlabcentral/fileexchange/53250-filtering-of-an-image-in-frequency-domain

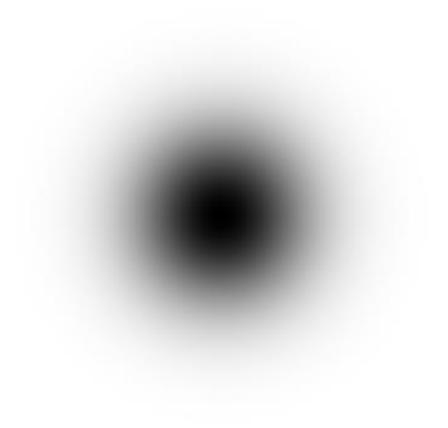
original image

padded image

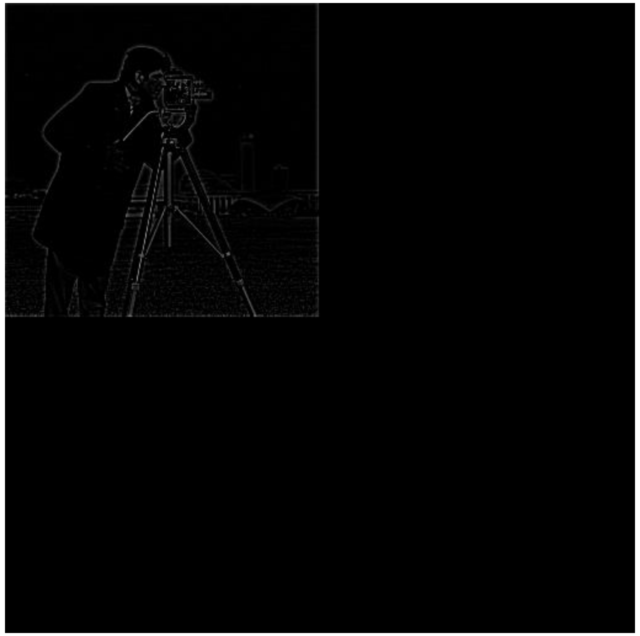
Fourier spectrum



Gaussian High Pas Filter

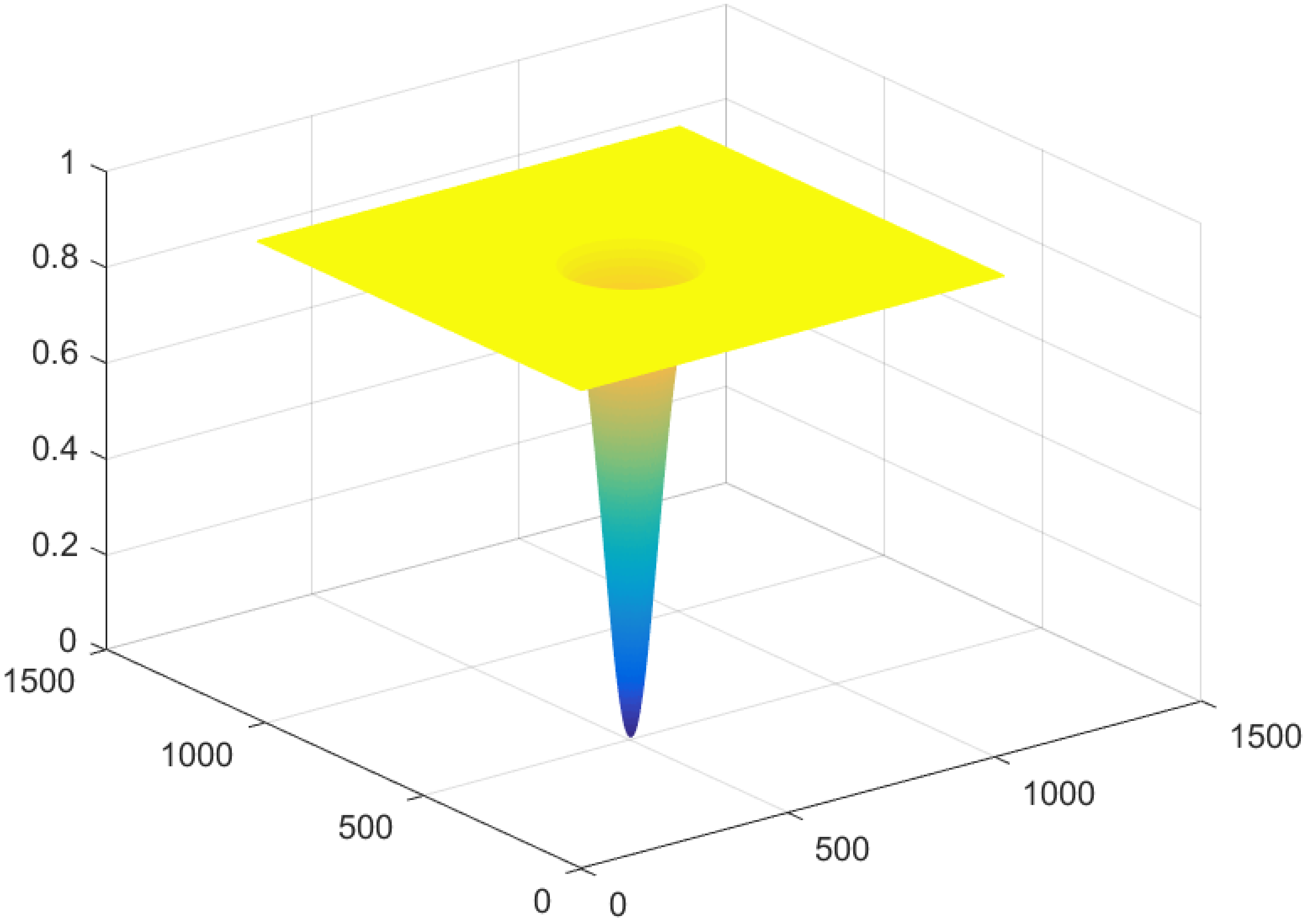


output image after inverse 2D DFT



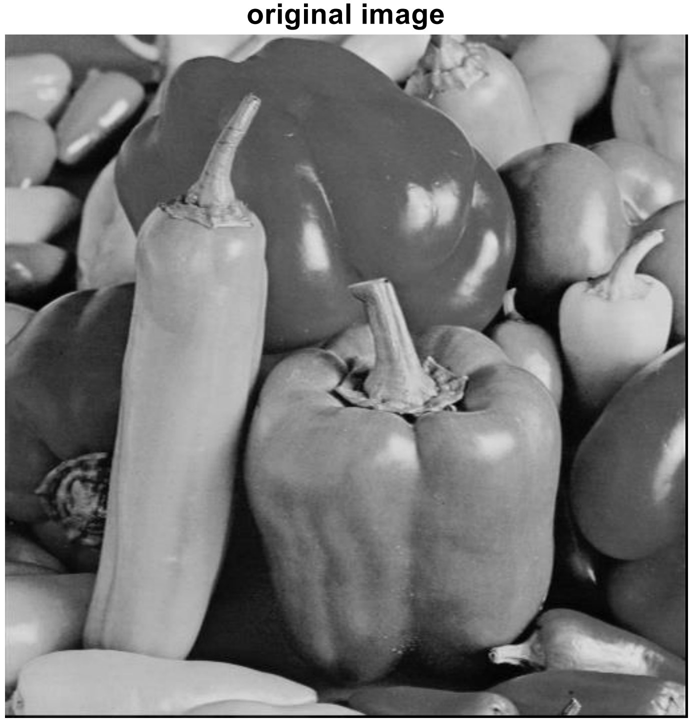
output image

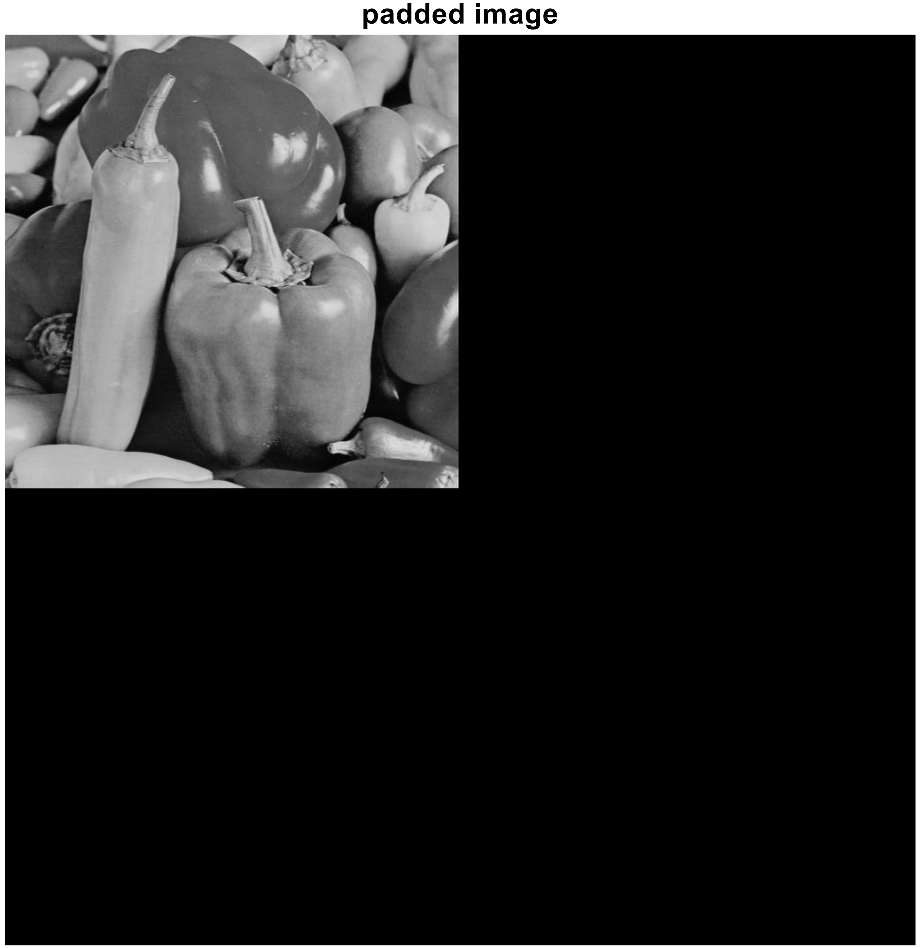
Hasil run program
dengan GHPF, D0 = 50



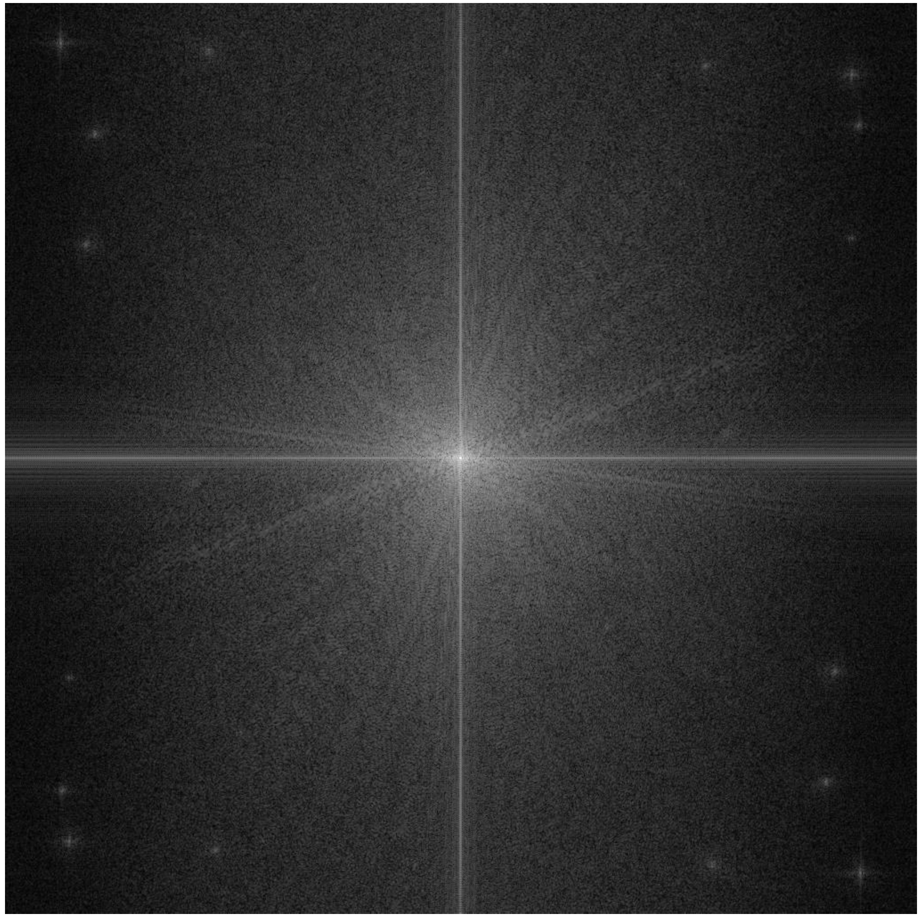
mesh(H)

Contoh hasil lain dengan citra *pepper*:





Fourier spectrum

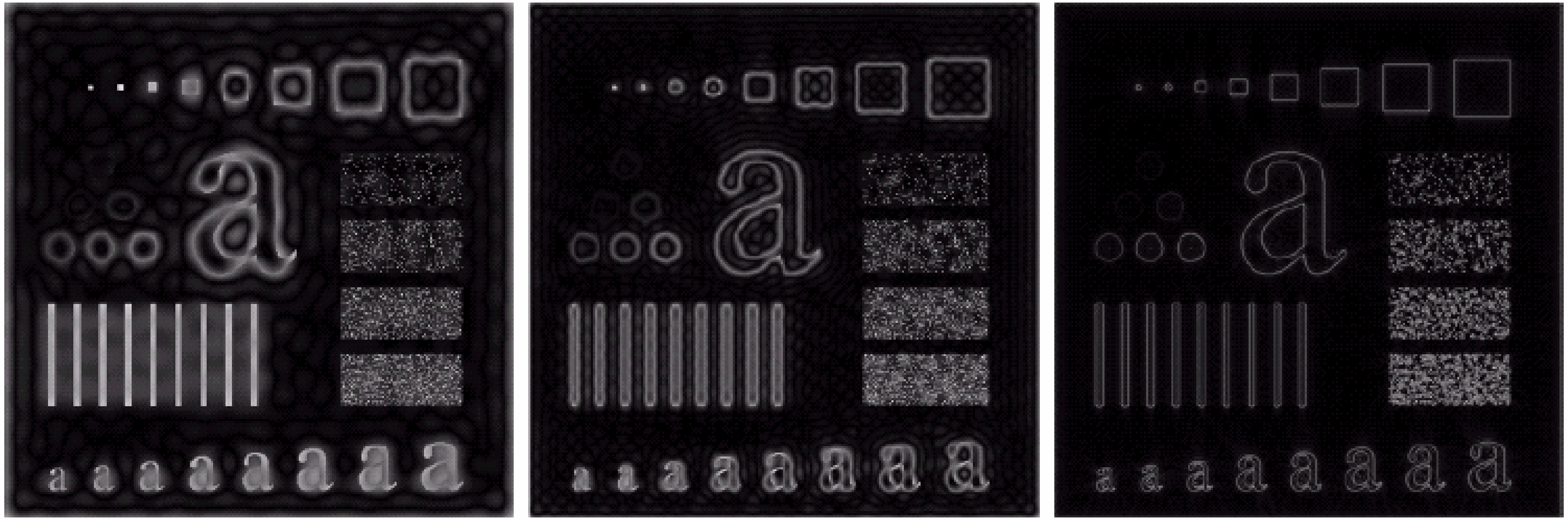


Gaussian High Pas Filter

output image after inverse 2D DFT

output image

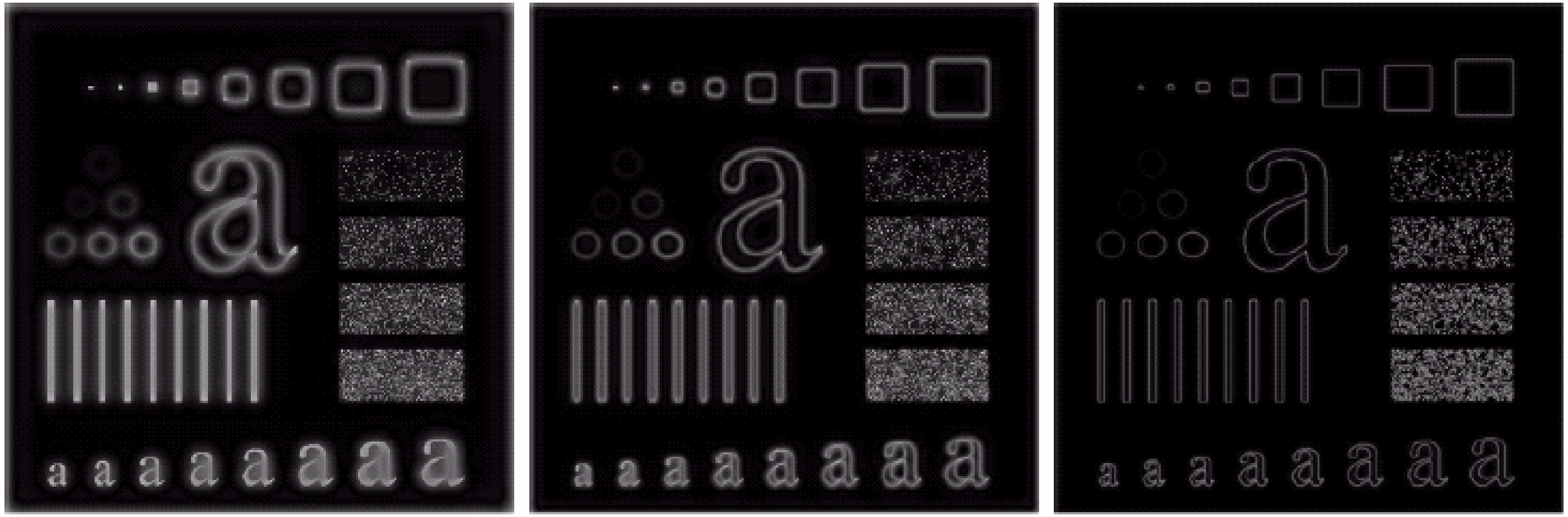
Example: result of IHPF



a b c

FIGURE 4.24 Results of ideal highpass filtering the image in Fig. 4.11(a) with $D_0 = 15$, 30, and 80, respectively. Problems with ringing are quite evident in (a) and (b).

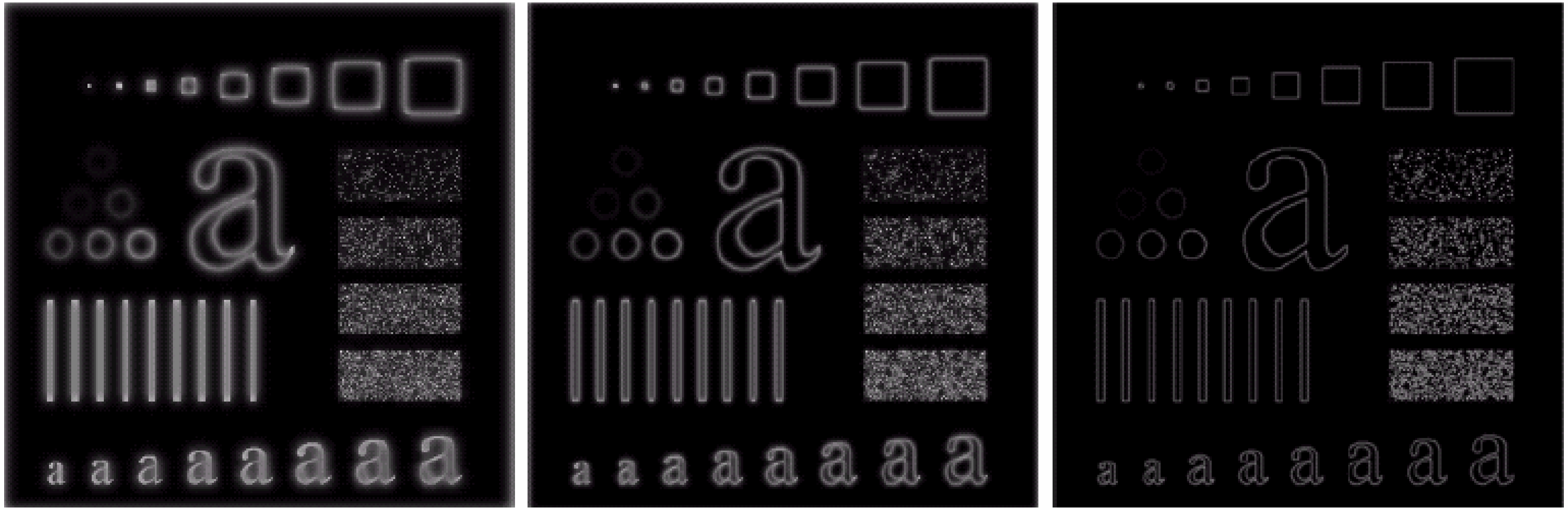
Example: result of BHPF



a b c

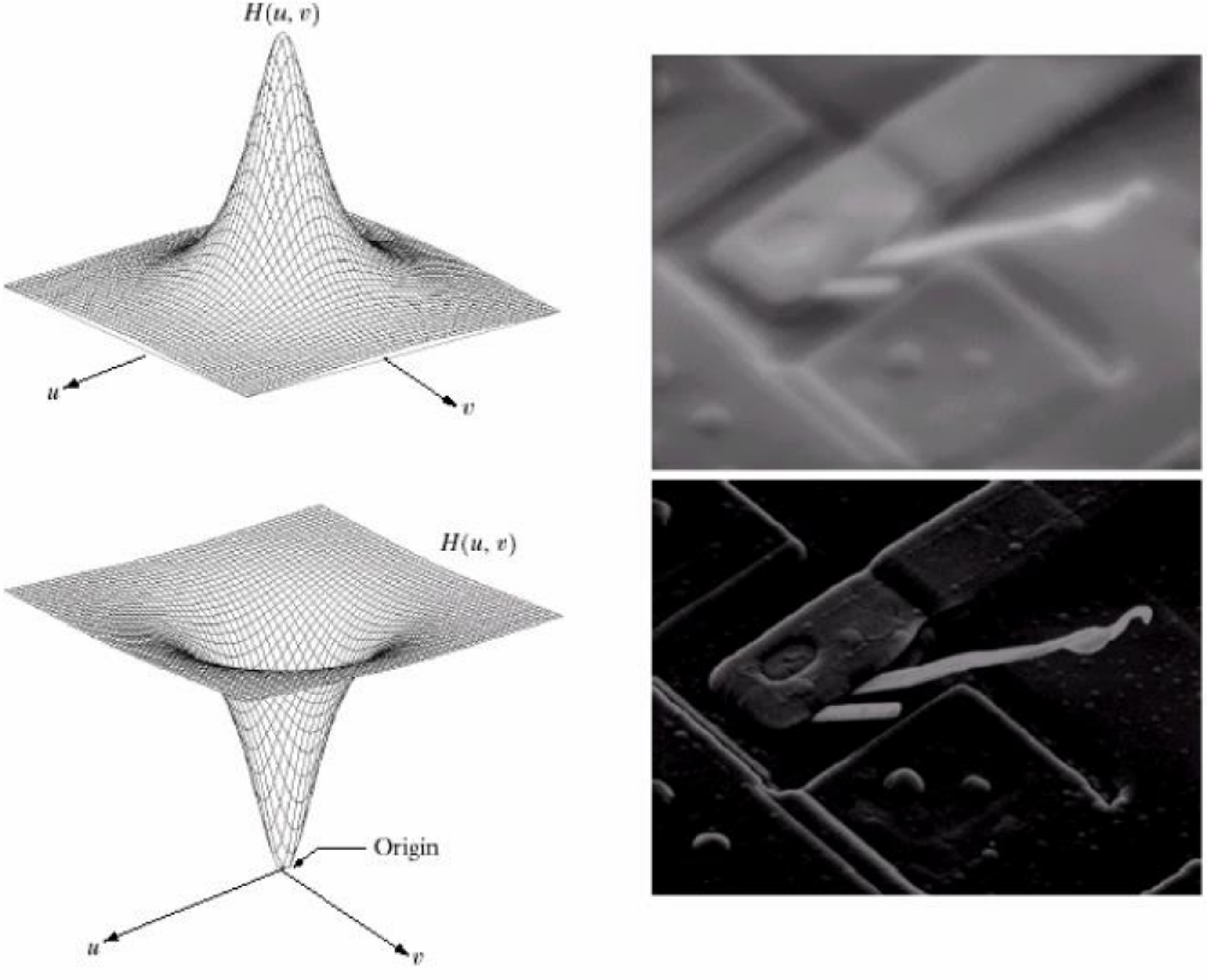
FIGURE 4.25 Results of highpass filtering the image in Fig. 4.11(a) using a BHPF of order 2 with $D_0 = 15$, 30, and 80, respectively. These results are much smoother than those obtained with an ILPF.

Example: result of GHPF



a b c

FIGURE 4.26 Results of highpass filtering the image of Fig. 4.11(a) using a GHPF of order 2 with $D_0 = 15$, 30, and 80, respectively. Compare with Figs. 4.24 and 4.25.



Low pass filter

High pass filter

a b
c d

FIGURE 4.7 (a) A two-dimensional lowpass filter function. (b) Result of lowpass filtering the image in Fig. 4.4(a). (c) A two-dimensional highpass filter function. (d) Result of highpass filtering the image in Fig. 4.4(a).

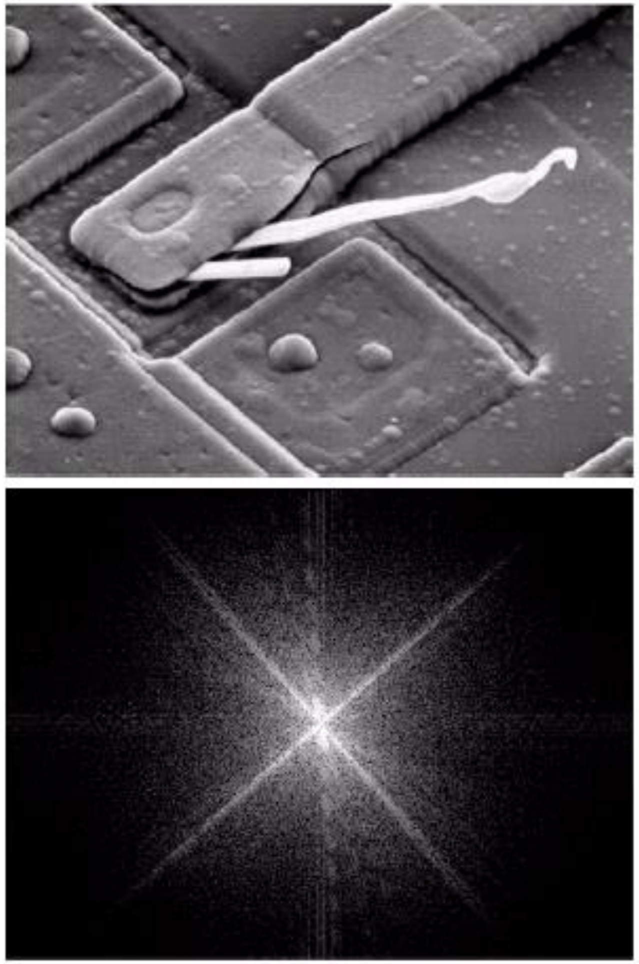
Penapis lubang (*notch filter*)

$$H(u,v) = \begin{cases} 0 & if\ (u,v) = (M/2, N/2) \\ 1 & otherwise \end{cases}$$

Note: F(0,0) is equal to the average value of f(x,y)

$$F(0,0) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y),$$

F(0,0) adalah pada lokasi u = M/2 dan v = N/2



a
b

FIGURE 4.4
(a) SEM image of a damaged integrated circuit. (b) Fourier spectrum of (a). (Original image courtesy of Dr. J. M. Hudak, Brockhouse Institute for Materials Research, McMaster University, Hamilton, Ontario, Canada.)

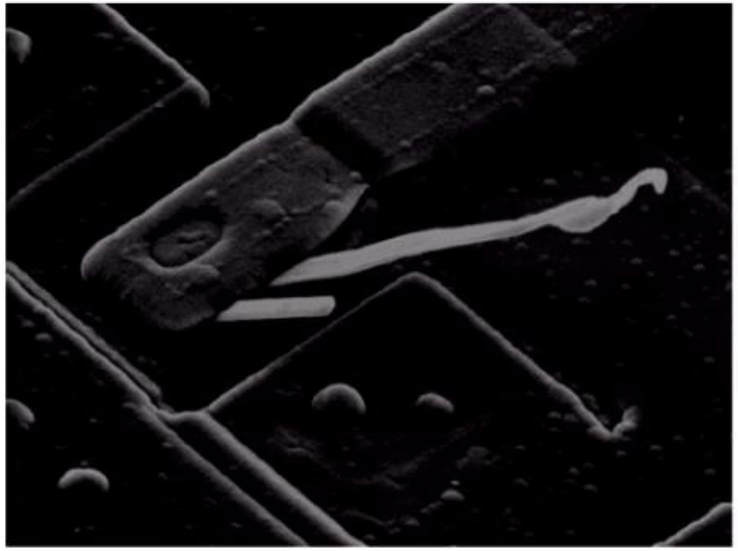


FIGURE 4.6
Result of filtering the image in Fig. 4.4(a) with a notch filter that set to 0 the $F(0, 0)$ term in the Fourier transform.

Unsharp Masking dan High-Boost Filtering dalam Ranah Frekuensi

- **Unsharp Masking**: $f_s(x,y) = f(x,y) - f_{lp}(x,y)$

$$F_s(u,v) = F(u,v) - F_{lp}(u,v) = \underbrace{(1 - H_{lp}(u,v))}_{H_{hp}(u,v)}F(u,v)$$

- **High-Boost Filtering**: $f_{hb}(x,y) = Af(x,y) - f_{lp}(x,y)$

$$(A \geq 1)$$

$$F_{hb}(u,v) = AF(u,v) - F_{lp}(u,v) =$$

$$= (A - H_{lp}(u,v))F(u,v) = (A - 1 + H_{hp}(u,v))F(u,v)$$